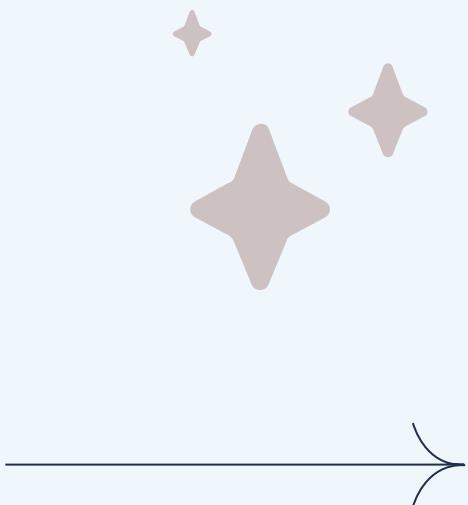


# “Balsam” For

Medical Devices.





## TABLE OF CONTENTS

01 ✧ Chapter 1

09 ✧ Chapter 2

10 Chapter 3



# TABLE OF CONTENTS

11

Chapter 4

12

Chapter 5

# Team member

Khadeejh Abdullah Gheliwi ID:443000079

Dai Ali Alqurashi ID:443001661

Shimaa Mohammed Al-khuzae ID:443000518

Jana Mohammed Medhar ID:442000261

Ghadah Ahmed Alharthi ID:443002768

Software Engineering 2

Dr. Rawan munshi



# Chapter 1

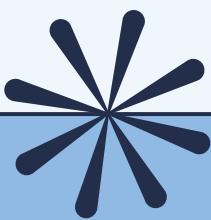
(Requirement Specification phase)





# Introducing the Balsam Project:

Revolutionizing Healthcare Access and Support.



The Balsam Project is an innovative initiative addressing challenges in the healthcare industry. It creates a holistic healthcare community and platform to improve accessibility and user experience.

- By bridging gaps in device access, information, and support, Balsam enhances well-being.
- The platform connects sellers, volunteers, renters, and beneficiaries for a comprehensive healthcare ecosystem.
- Services include product listing, technical support, device consultations, and volunteer opportunities.
- Balsam aims to revolutionize healthcare by providing convenient access and support for all.



# User and system

## Functional Requirement :

### 1. The User is able to create an account .

1.1:

- the first thing the user should do is to insert his information : email, phone number , password, username , full name .

1.2

- the user must chose weather he is a seller , a beneficiary , a donor or a renter.

1.3

- the user must click the icon “create account “ when he finishes .  
- the information will be saved.

# User and system

## Functional Requirement :

### 2. The User Can login in the system

2.1:

-the user clicks the “log in “ icon

2.2

-the user must insert the username and the password.

2.3

- if the information matches the data that was save then it well proceed the login.

# User and system

## Functional Requirement :

### 3. The User shall browse, search and filter the products .

3.1:

- the user can scroll over the screen to watch all the products.

3.2

- the user can search about a specific item when he clicks on the searching icon and writes the item's name.

3.3

-the user can filleter the product according to his specifications by clicking on the filter icon.

## Functional Requirement :

### 4. The User can review and rate the products

4.1:

- under each product there will be a review and rate for the product.

4.2

- the users can write their opinion after using the service.

4.3

- the user can rate the service

# User and system

## Functional Requirement :

5. The system provide technical support to users  
for problem resolution and inquiries.

5.1:

-when the user have inquiries or a technical issues he shall click on the technical support icon and get the help he wants.

5.2

- The technical support request interface shall include fields for the user to enter their name, contact information, a detailed description of the problem or inquiry, and any relevant attachments or screenshots.

5.3

- When a support request is marked as “Resolved,” the system shall send an automated notification to the user, informing them that their issue has been resolved and providing any necessary instructions or follow-up actions.

# User and system

## Functional Requirement:

6. The system offer medical advice or consultations about the provided devices.

### 6.1

-The system should enable users to designate a product as a favorite by clicking the star icon displayed with the device information.



### 6.2

-The selected product will be added to the user's favorites list.



# User and system

## Functional Requirement :

7) The system provide technical support to users for problem resolution and inquiries.

### 6.1

-When users require consultations regarding device functionality or any medical issue, they can seek assistance by submitting questions through the "What is your question" field.



### 6.2

-Any available consultant will contact the user to provide assistance, answering questions, and explaining device functionality or addressing medical concerns.



# User and system

## Functional Requirement :

8) The Users have the option to request delivery and installation services for purchased devices.

### 8.1

-Following the selection of a device and completion of necessary information, the system must present users with two optional services.

### 8.2

-delivery and installation. For the delivery service, users must input their address

### 8.3

-The installation service's availability should be determined based on the specific requirements of the chosen device, indicating whether expert installation is necessary

# Non-functional Requirement:

## Dependability:

- 
- 
- 
- 

The maximum allowable downtime for a system, or application following a failure or disaster is three hours.

## Usability:

- 
- 
- 
- 

The system shall provide a simple, intuitive interface that facilitates the all users navigation on the app.

## Scalability:

- 
- 
- 
- 

The system architecture should be designed to scale seamlessly, accommodating a growing user base and increased service demands without compromising performance.

## Security:

- 
- 
- 
- 

The platform must implement stringent security measures, including two-step verification and data encryption, to safeguard user information and transactions effectively.

# System models



02

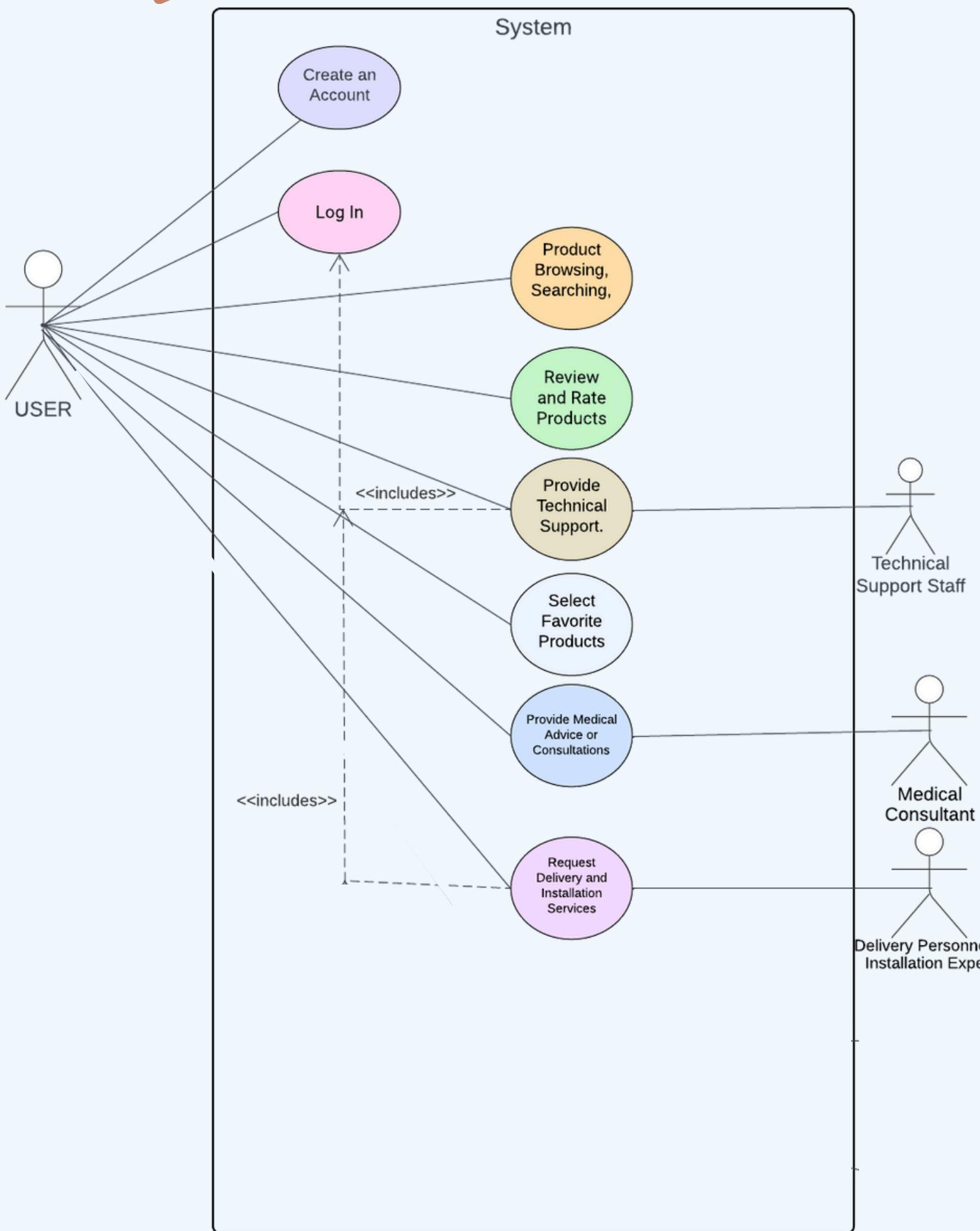
# ~~USE CASE AND SCENARIO:~~



03



# USE CASE Diagram:





## 1. Use Case: User Account Creation.

<b>Actors:</b>	User, System
<b>Description:</b>	A user may create an account in the system. The information provided includes personal details and the user's role. Once the account is created, the information is saved in the system.
<b>Data:</b>	<ul style="list-style-type: none"><li>- User information: email, phone number, password, username, full name.</li><li>- User role: seller, beneficiary, donor, renter.</li></ul>
<b>Stimulus:</b>	The user initiates the account creation process by clicks the "Create Account" button or link.
<b>Response:</b>	<ol style="list-style-type: none"><li>1. System displays a form for user information entry.</li><li>2. User enters their information and selects their role.</li><li>3. User clicks the "Create Account" button to submit the form.</li><li>4. System validates the information.</li><li>5. If information is valid:<ul style="list-style-type: none"><li>- System creates the account.</li><li>- System sends a confirmation email to the user.</li><li>- System displays a confirmation message to the user.</li></ul></li><li>6. If information is invalid:<ul style="list-style-type: none"><li>- System displays error messages for invalid fields.</li></ul></li></ol>
<b>Comment:</b>	<p>The system must:</p> <ul style="list-style-type: none"><li>- Ensure unique email and username.</li><li>- Securely store and validate passwords and enforce any necessary validation rules, such as password complexity requirements or email format validation.</li><li>- Provide user feedback on account creation is successful or if there are any errors .</li></ul>



## 2. Use Case: User Login.

<b>Actors:</b>	User, System
<b>Description:</b>	A user may log into the system using their username and password. The system verifies the information and proceeds with the login if the information matches the saved data
<b>Data:</b>	<ul style="list-style-type: none"><li>- Username: User's unique identifier.</li><li>- Password: User's secret authentication credential.</li></ul>
<b>Stimulus:</b>	The user initiates the login process by clicking the "Log In" icon or button on the system's login page.
<b>Response:</b>	<ol style="list-style-type: none"><li>1. The user clicks the "log in" icon.</li><li>2. The system prompts the user to enter their username and password.</li><li>3. The user enters their username and password.</li><li>4. The system verifies the information.</li><li>5. If the information matches the saved data, the system proceeds with the login.</li></ol>
<b>Comment:</b>	<p>The system must:</p> <ul style="list-style-type: none"><li>- Securely verify passwords.</li><li>- Provide feedback on login status (such as incorrect username or password).</li><li>- Implement security measures for multiple failed logins, such as temporarily locking the account or enforcing a password reset.</li><li>- Grant access to account and features upon successful login.</li></ul>



### 3. Use Case: Browse, Search, and Filter Products.

<b>Actors:</b>	User, System
<b>Description:</b>	A user may browse, search, and filter the products available in the system. The user can scroll over the screen to view all products, search for a specific item, and filter products according to their specifications.
<b>Data:</b>	<ul style="list-style-type: none"><li>-Product information (name, description, category, specifications, etc.)</li><li>-User search terms.</li><li>-User-specified filtering criteria.</li></ul>
<b>Stimulus:</b>	The user interacts with the system by scrolling, searching, and applying filters.
<b>Response:</b>	<ol style="list-style-type: none"><li>1. System displays a list of available products.</li><li>2. User may scroll through the list to view products.</li><li>3. User may initiate a search:<ul style="list-style-type: none"><li>- User clicks the "Search" icon.</li><li>- User enters search terms.</li><li>- System displays products matching the search terms.</li></ul></li><li>4. User may apply filters:<ul style="list-style-type: none"><li>- User clicks the "Filter" icon.</li><li>- User selects desired filtering criteria.</li><li>- System refines the product list based on the filters.</li></ul></li></ol>
<b>Comment:</b>	<p>The system should:</p> <ul style="list-style-type: none"><li>- Provide user-friendly interfaces for scrolling, searching, and filtering.</li><li>- Ensure search and filter results are accurate and relevant.</li><li>- Design filter options based on product attributes and user preferences.</li><li>- Provide feedback if no products match the search or filter criteria.</li></ul>



## 4. Use Case: Review and Rate Products.

<b>Actors:</b>	User, System
<b>Description:</b>	A user may review and rate the products available in the system. Under each product, there will be a section for reviews and ratings. Users can write their opinion after using the service and rate the service.
<b>Data:</b>	<ul style="list-style-type: none"><li>- Product ID</li><li>- User ID</li><li>- Review text</li><li>- Rating (e.g., 1-5 stars)</li><li>- Date and time of review</li></ul>
<b>Stimulus:</b>	<ul style="list-style-type: none"><li>- User views a product page and decides to leave a review or rating.</li></ul>
<b>Response:</b>	<ol style="list-style-type: none"><li>1. System prompts user to enter their review text and rating.</li><li>2. User submits their feedback.</li><li>3. System validates and stores the review and rating in the database.</li><li>4. System displays the new review and rating on the product page, along with an average rating calculated from all reviews.</li></ol>
<b>Comment:</b>	<p>The system should:</p> <ul style="list-style-type: none"><li>- Encourage users to leave reviews, possibly through rewards or highlighting top reviewers.</li><li>- Recognize that user reviews and ratings contribute to product reputation or ranking.</li><li>- Implement moderation mechanisms to prevent spam or inappropriate content in reviews.</li><li>- Ensure only users who have used a product can review and rate it.</li></ul>



## 5.Use Case: Provide Technical Support.

<b>Actors:</b>	User, System,( Technical Support Staff )
<b>Description:</b>	The system allows users to submit technical support requests for problem resolution or inquiries, providing assistance to ensure smooth system usage.
<b>Data:</b>	<ul style="list-style-type: none"><li>- User name</li><li>- Contact information (email, phone number)</li><li>- Problem/inquiry description</li><li>- Attachments (screenshots, logs, etc.)</li><li>- Support request status (e.g., Open, Pending, Resolved)</li><li>- Resolution details (if applicable)</li></ul>
<b>Stimulus:</b>	-User encounters a technical issue or has a question and clicks the "Technical Support" icon.
<b>Response:</b>	<ol style="list-style-type: none"><li>1. System displays support request form.</li><li>2. User submits completed form with required information and attachments.</li><li>3. Technical support staff reviews, troubleshoots, and provides resolution.</li><li>4. System marks request as "Resolved" and notifies user.</li><li>5. User receives notification and follows provided instructions or actions.</li></ol>
<b>Comment:</b>	<p>The system should:</p> <ul style="list-style-type: none"><li>- Have a user-friendly interface with clear instructions for submitting support requests.</li><li>- Ensure the security and confidentiality of user's contact information and attached files.</li><li>- Have an efficient tracking system for support requests.</li><li>- Require valid information for support request submission.</li><li>- Provide appropriate error messages for invalid or incomplete information.</li></ul>



## 6. Use Case: Select Favorite Products.

<b>Actors:</b>	User, System.
<b>Description:</b>	Users can mark products as favorites for easy access and recall, creating a personalized collection of preferred items.
<b>Data:</b>	<ul style="list-style-type: none"><li>- User ID</li><li>- Product ID</li><li>- Favorite status (true/false)</li></ul>
<b>Stimulus:</b>	<ul style="list-style-type: none"><li>- User views a product and decides to add it to their favorites by clicking the "star" icon.</li></ul>
<b>Response:</b>	<ol style="list-style-type: none"><li>1. User clicks the "star" icon associated with the product.</li><li>2. System records the product as a favorite for the user.</li><li>3. System visually indicates the favorite status (e.g., highlights the star icon, adds product to a favorites list).</li></ol>
<b>Comment:</b>	<p>The system should:</p> <ul style="list-style-type: none"><li>- Provide intuitive indicators for users to mark products as favorites (such as a star icon).</li><li>- Make the favorites list accessible from user's profile, a dedicated page, or a personalized dashboard.</li><li>- Allow users to add or remove products from their favorites list.</li><li>- Integrate favorites functionality with features like personalized recommendations and search filters.</li></ul>



## 7. Use Case: Provide Medical Advice or Consultations.

<b>Actors:</b>	User, System , Medical Consultant.
<b>Description:</b>	Users can submit questions related to device functionality or medical concerns through the system, receiving assistance from medical consultants.
<b>Data:</b>	<ul style="list-style-type: none"><li>- User ID</li><li>- Device ID (optional)</li><li>- Question/concern description</li><li>- Medical history (optional)</li><li>- Consultant/representative response</li><li>- Consultation session data (e.g., date, time, duration)</li></ul>
<b>Stimulus:</b>	<ul style="list-style-type: none"><li>- User encounters a medical issue or has functional questions about their device, and decides to seek help by submitting a question through the "ASK US ! WE HERE TO HELP &lt;3" field.</li></ul>
<b>Response:</b>	<ol style="list-style-type: none"><li>1. User submits question or concern via "ASK US! WE'RE HERE TO HELP &lt;3" field.</li><li>2. System routes request to an available medical consultant.</li><li>3. Consultant contacts user through preferred communication channels(e.g., phone, text, video call).</li><li>4. Consultant provides clarification, explanation, and advice on device functionality.</li><li>5. Consultation session is documented and stored for future reference.</li></ol>
<b>Comment:</b>	<ul style="list-style-type: none"><li>- Conduct user-consultant communication securely, respecting privacy and confidentiality.</li><li>- Implement scheduling or appointment features if necessary.</li><li>- Ensure consultants provide accurate and reliable medical advice.</li><li>- Handle high volume of queries and ensure timely responses.</li></ul>



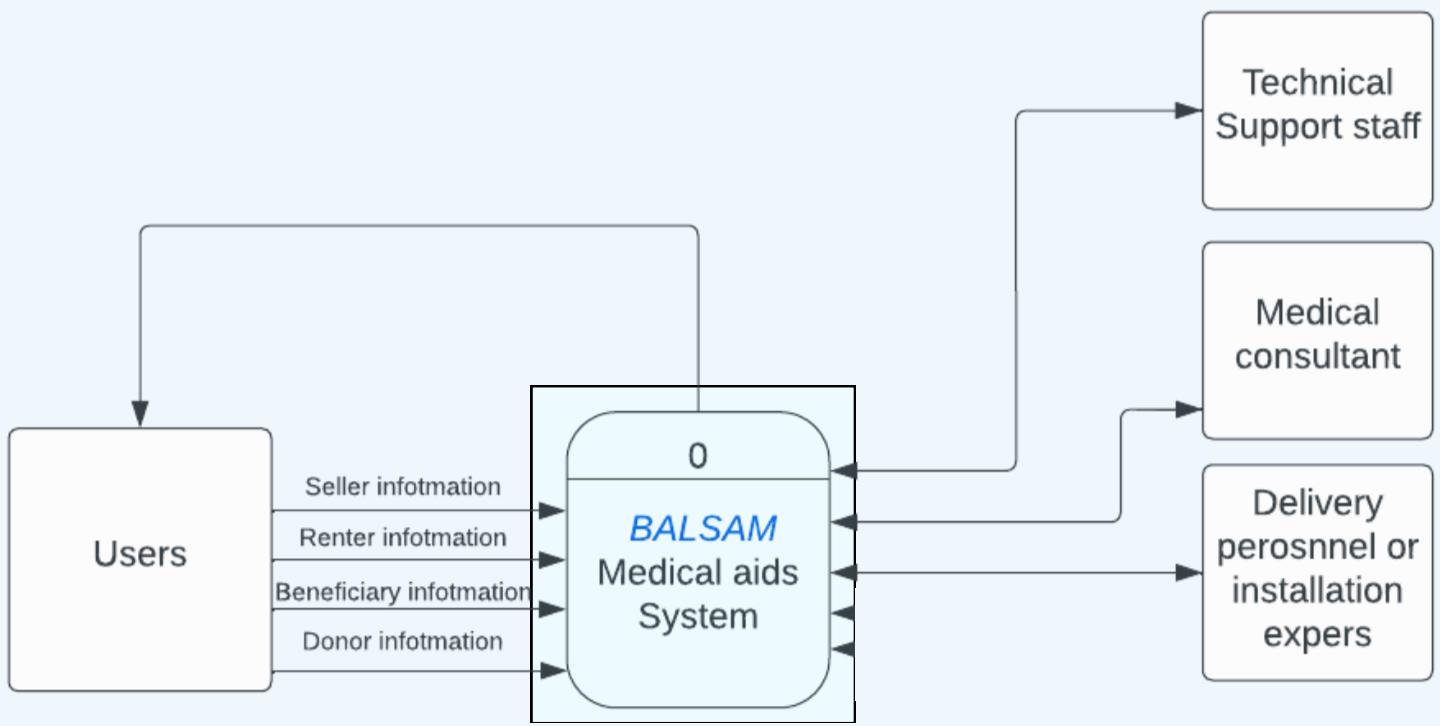
## 8. Use Case: Request Delivery and Installation Services.

<b>Actors:</b>	User, System, Delivery Personnel , Installation Expert
<b>Description:</b>	This use case describes the steps involved in the user's interaction with the system to request delivery and installation services for purchased devices. The user can select these optional services during the checkout process, providing their address for delivery and determining if expert installation is necessary based on the device's requirements.
<b>Data:</b>	<ul style="list-style-type: none"><li>- User ID</li><li>- Device ID</li><li>- Delivery address</li><li>- Installation requirements</li><li>- Service selection (delivery, installation, or both)</li><li>- Delivery schedule</li><li>- Installation appointment</li><li>- Delivery/installation status</li></ul>
<b>Stimulus:</b>	<ul style="list-style-type: none"><li>- User completes a device purchase and is presented request delivery and installation services for purchased devices.</li></ul>
<b>Response:</b>	<ol style="list-style-type: none"><li>1. Display device delivery and installation options, take delivery address.</li><li>2. Schedule delivery/installation based on availability and requirements.</li><li>3. Track and update delivery/installation progress.</li><li>4. Mark services as complete upon successful completion.</li></ol>
<b>Comment:</b>	<p>The system should:</p> <ol style="list-style-type: none"><li>1. Ensure user's personal information security and privacy.</li><li>2. Clearly communicate costs and integrate payment options.</li><li>3. Allow users to select preferred delivery/installation schedule.</li><li>4. Provide delivery/installation tracking information.</li><li>5. Collect user feedback for service improvement.</li></ol>

# 04

# Context diagram



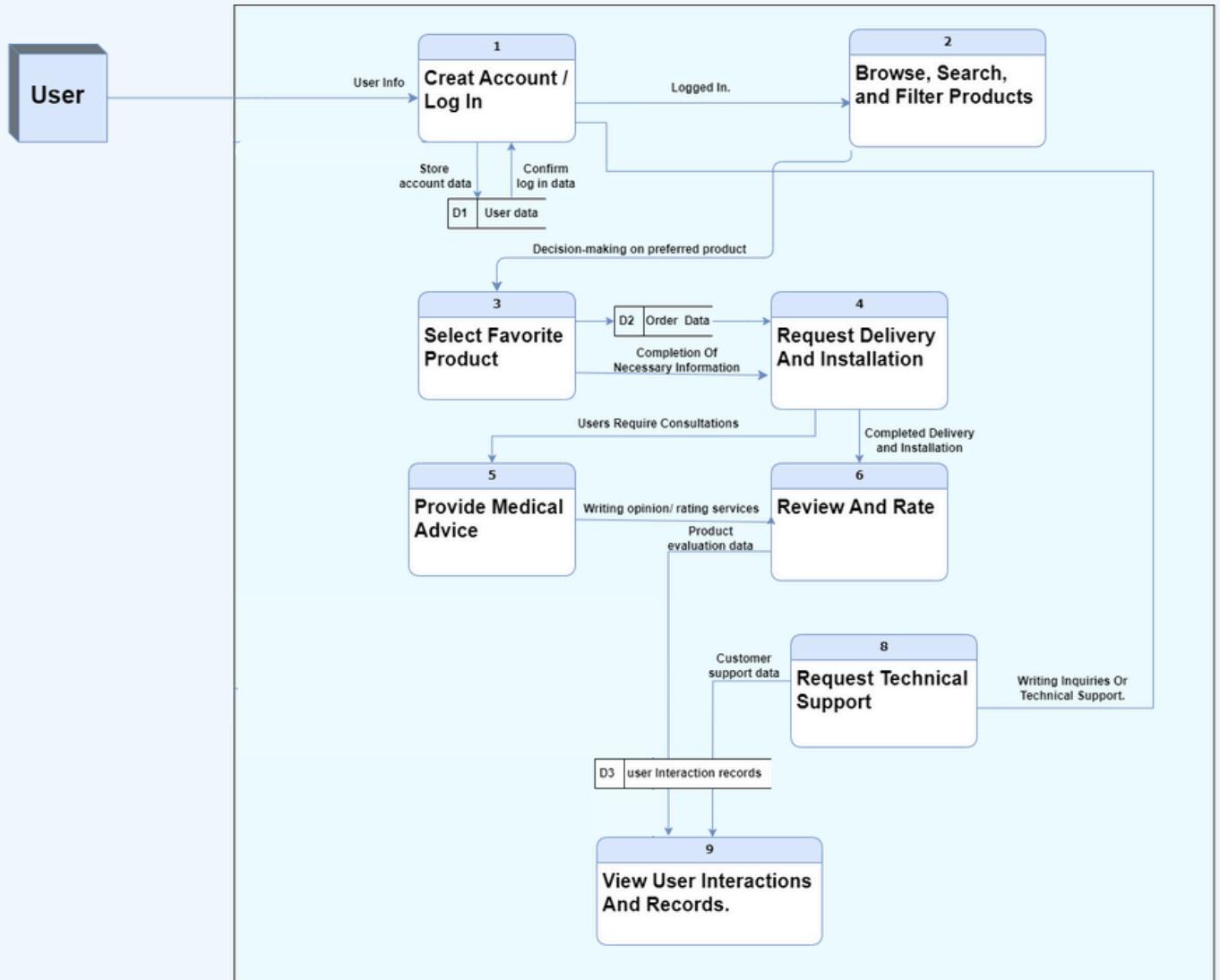


# 05

## DFD level 1

USE Case Scenario and The Diagram:

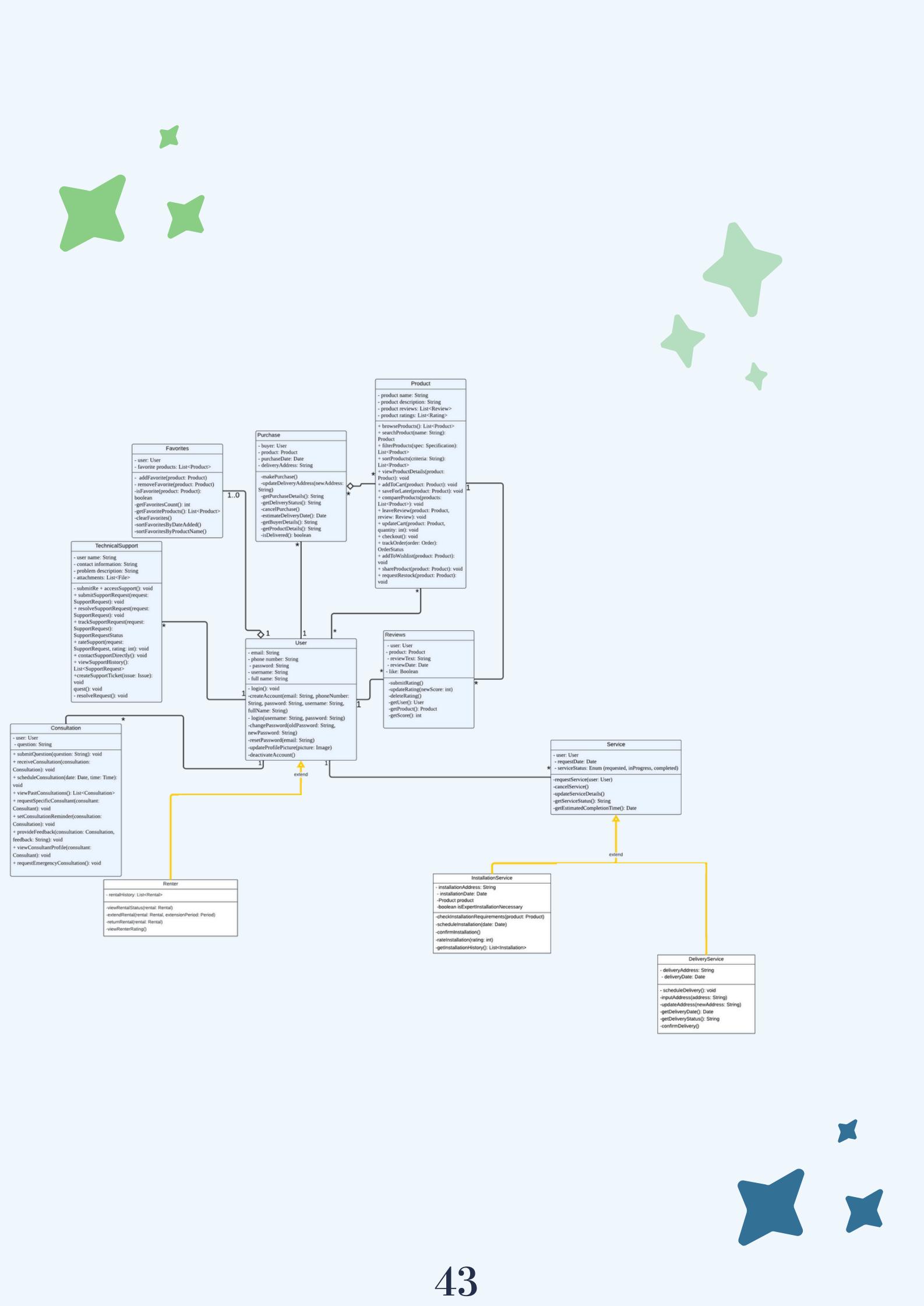






# 06

# Class Diagram



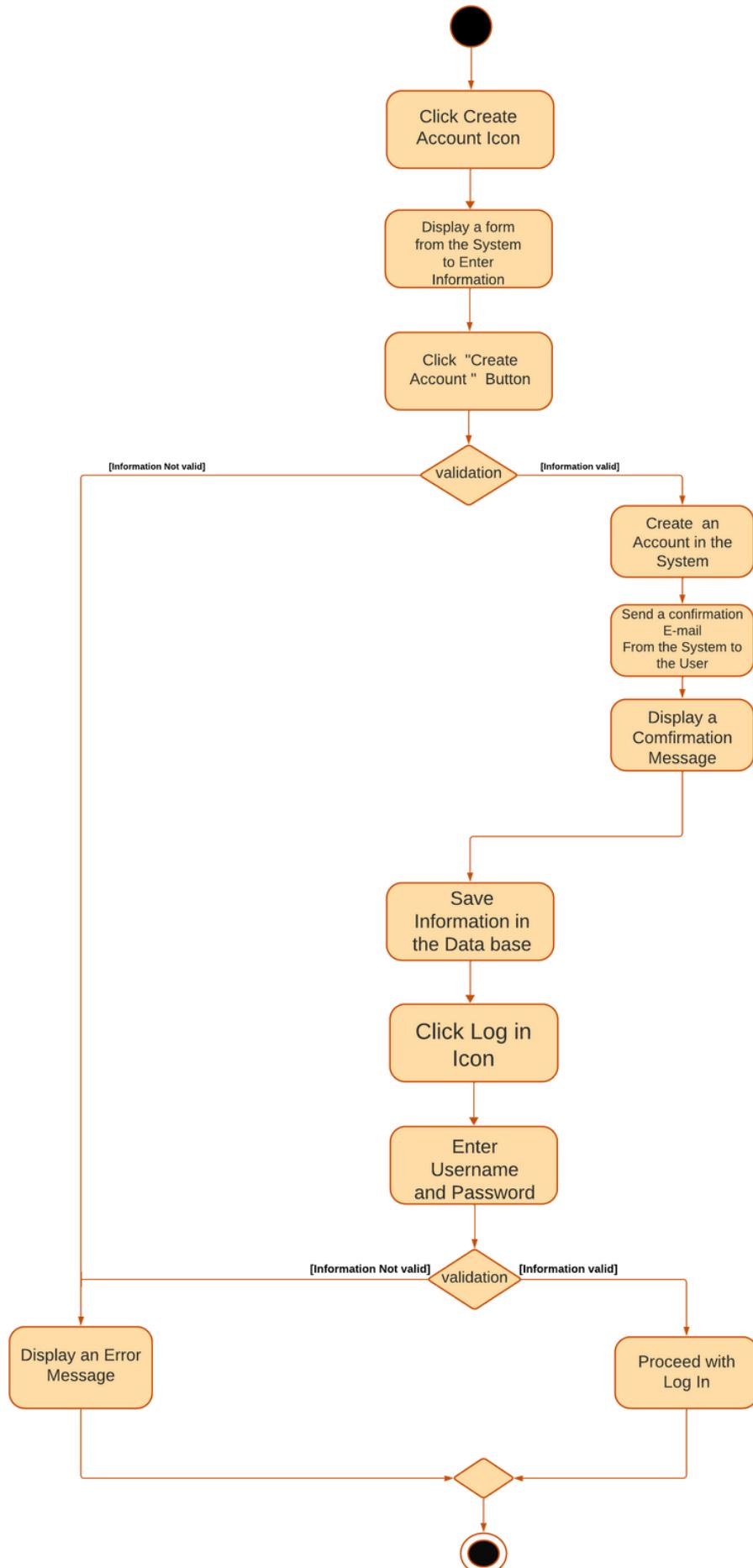
# 07

# Activity diagrams

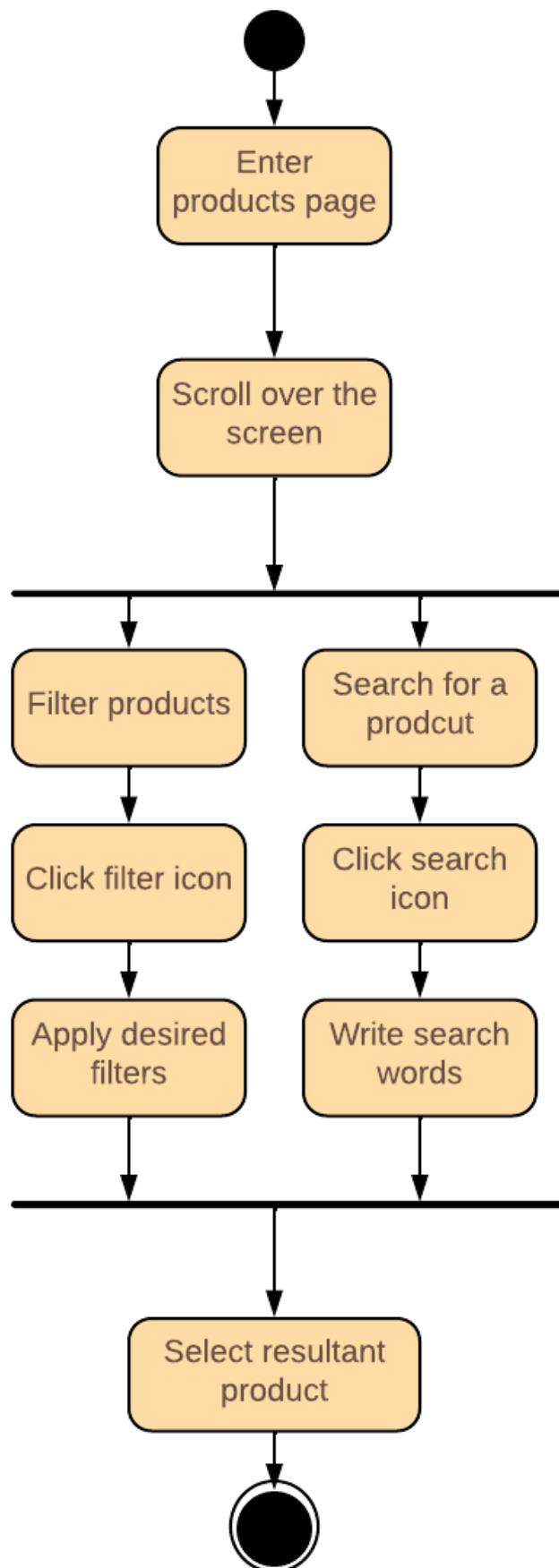
USE Case Scenario and The Diagram:



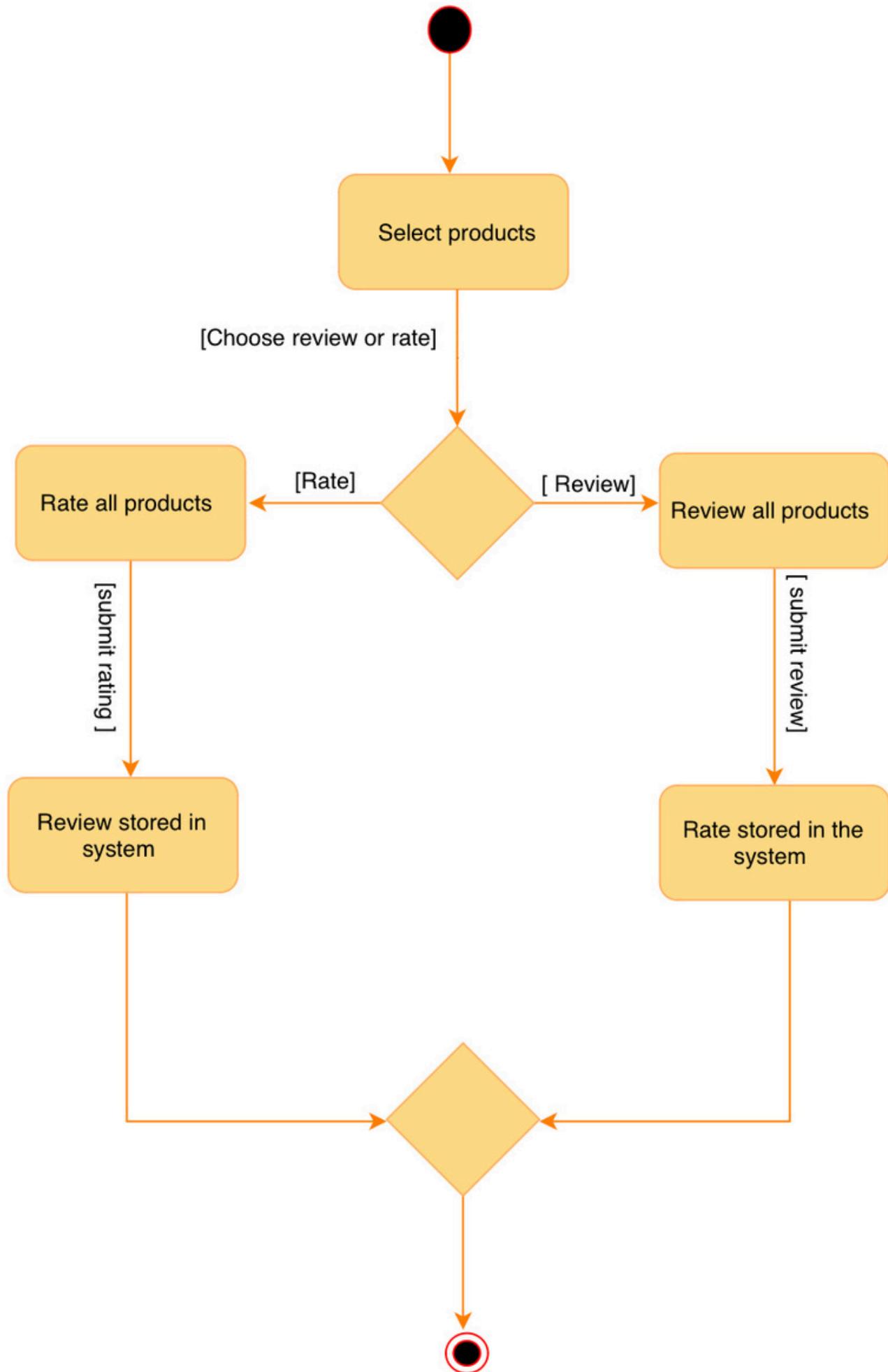
# Activity diagram for: Create account and Log in



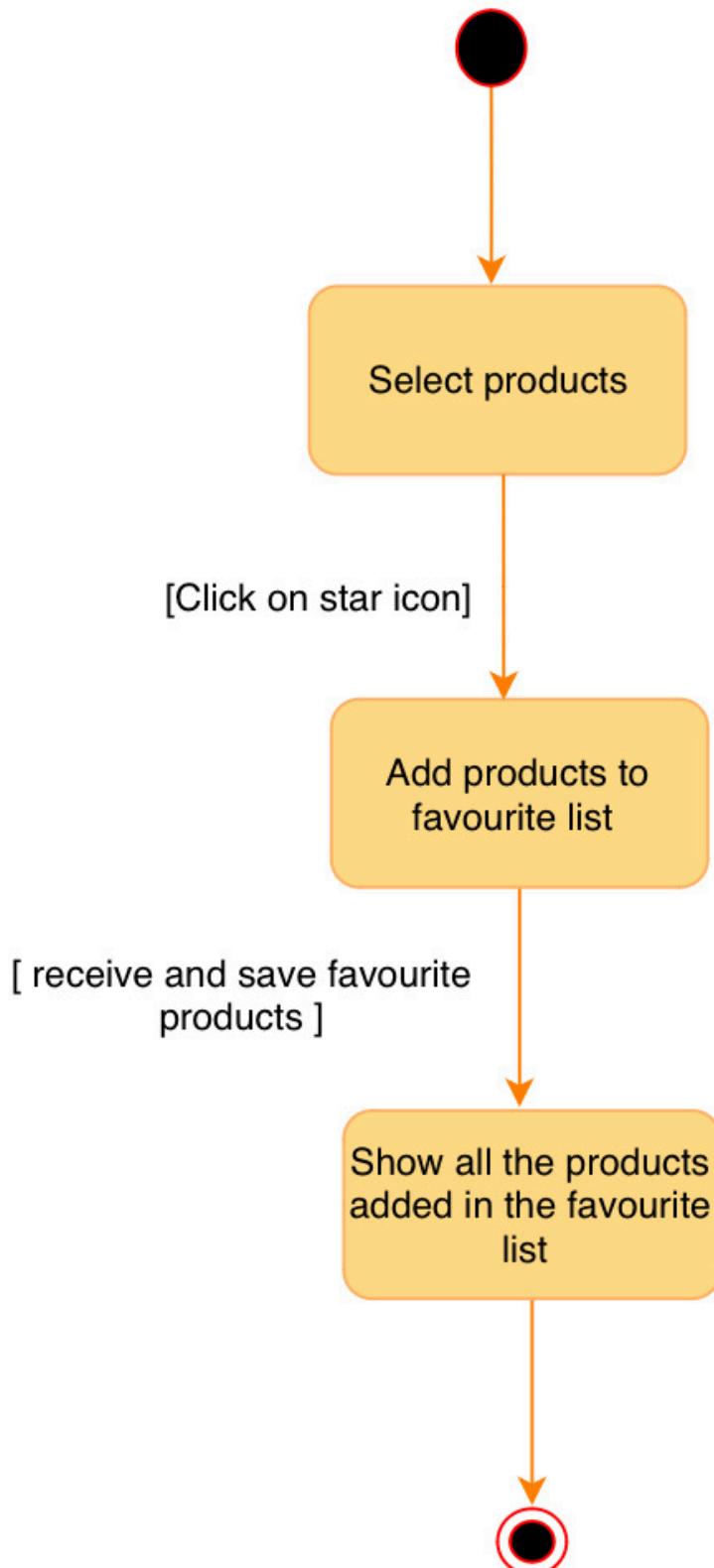
# Activity diagram for: Browse,search, and filter products



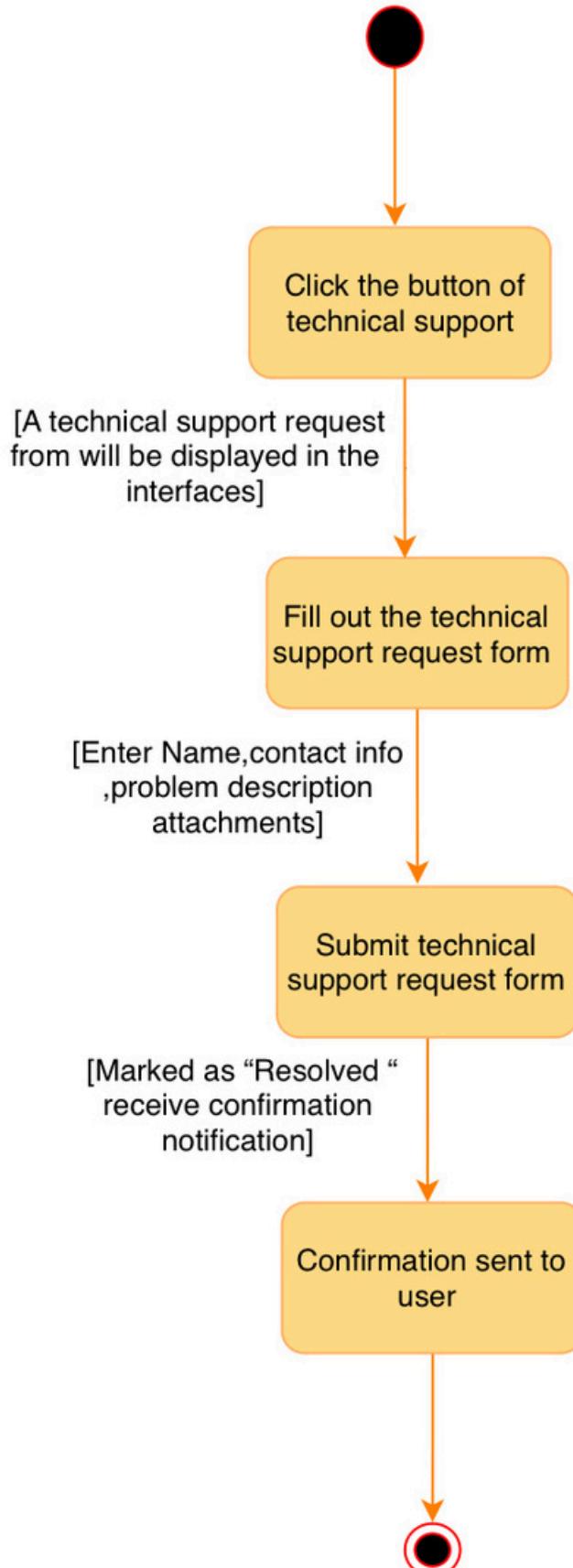
## Activity diagram for: Review and rate



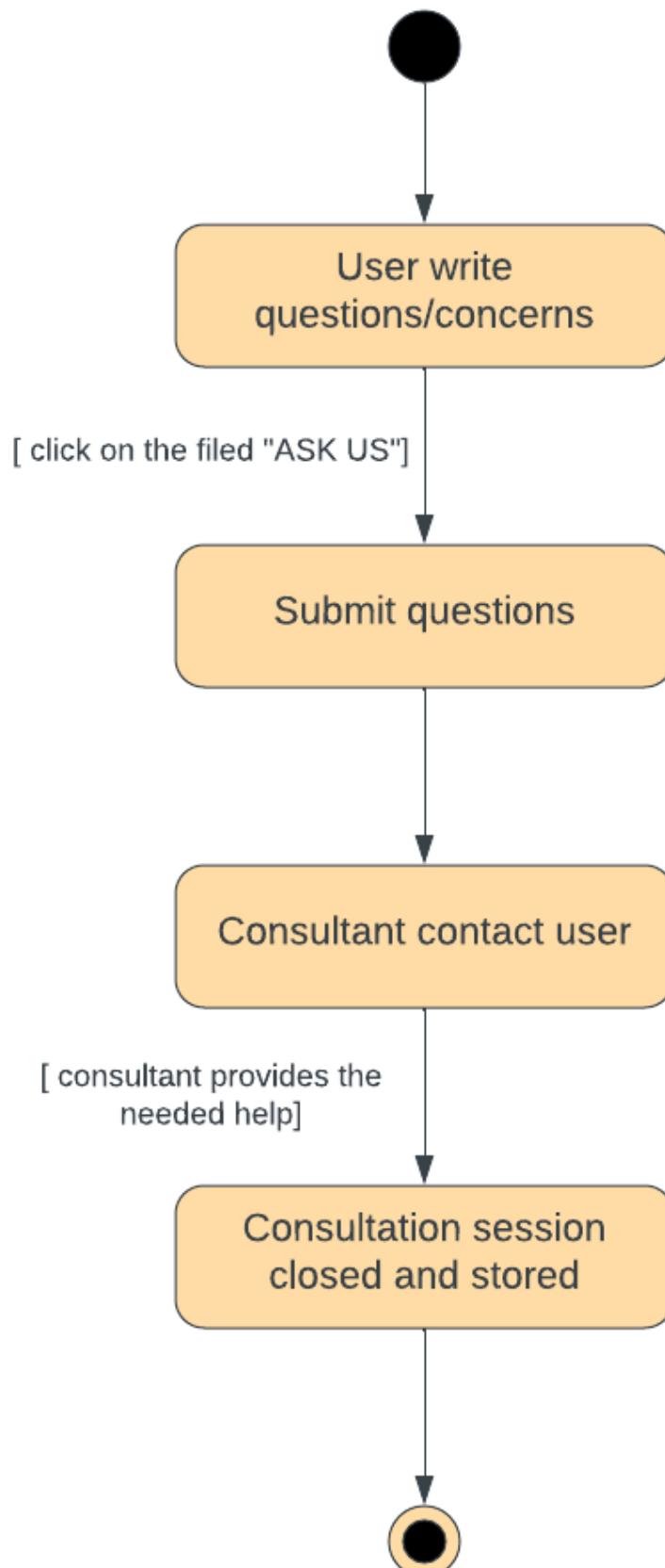
## Activity Diagram for select favourite products



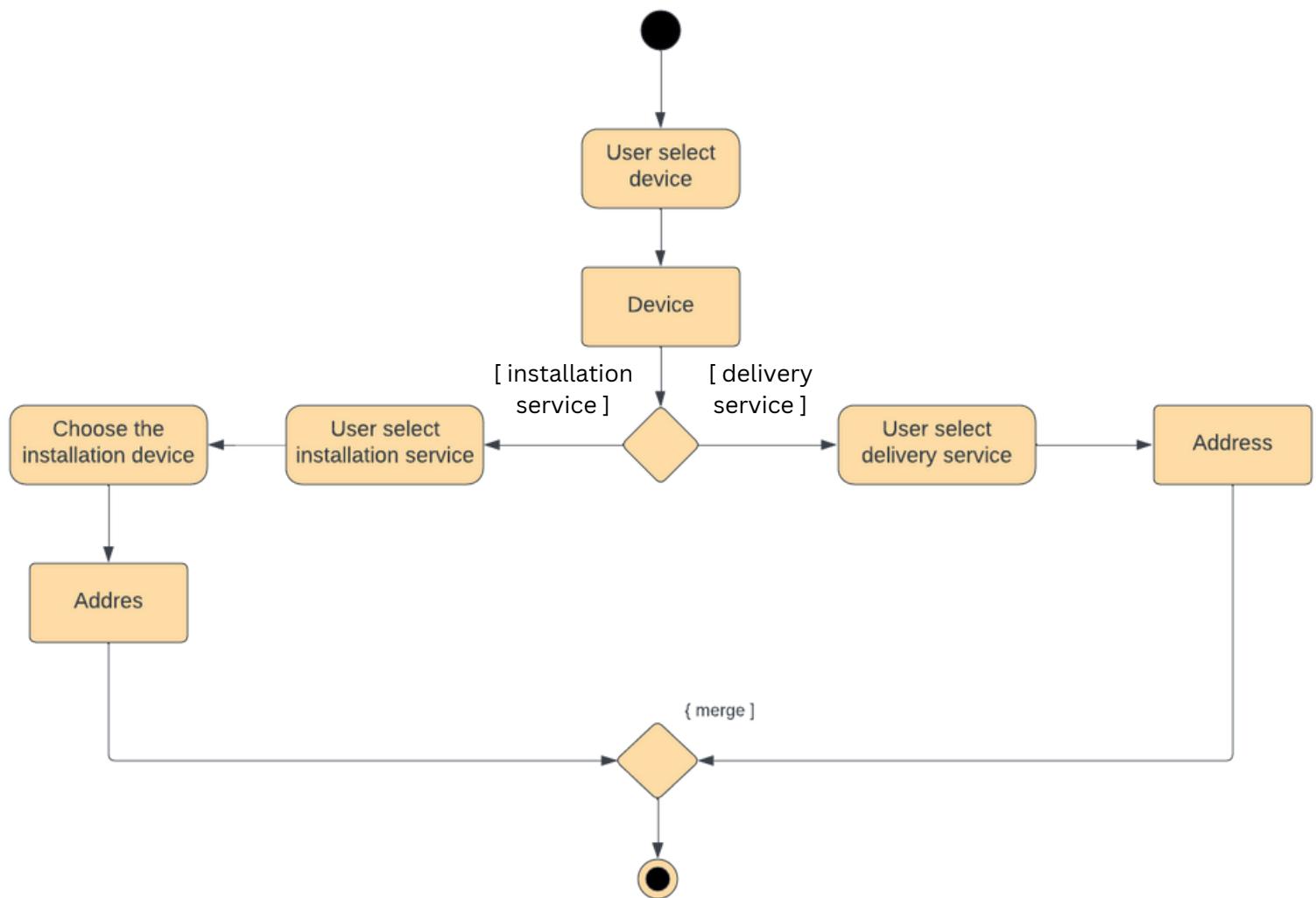
## Activity Diagram for provide technical support



# Activity diagram for: Provide medical advice or consultation



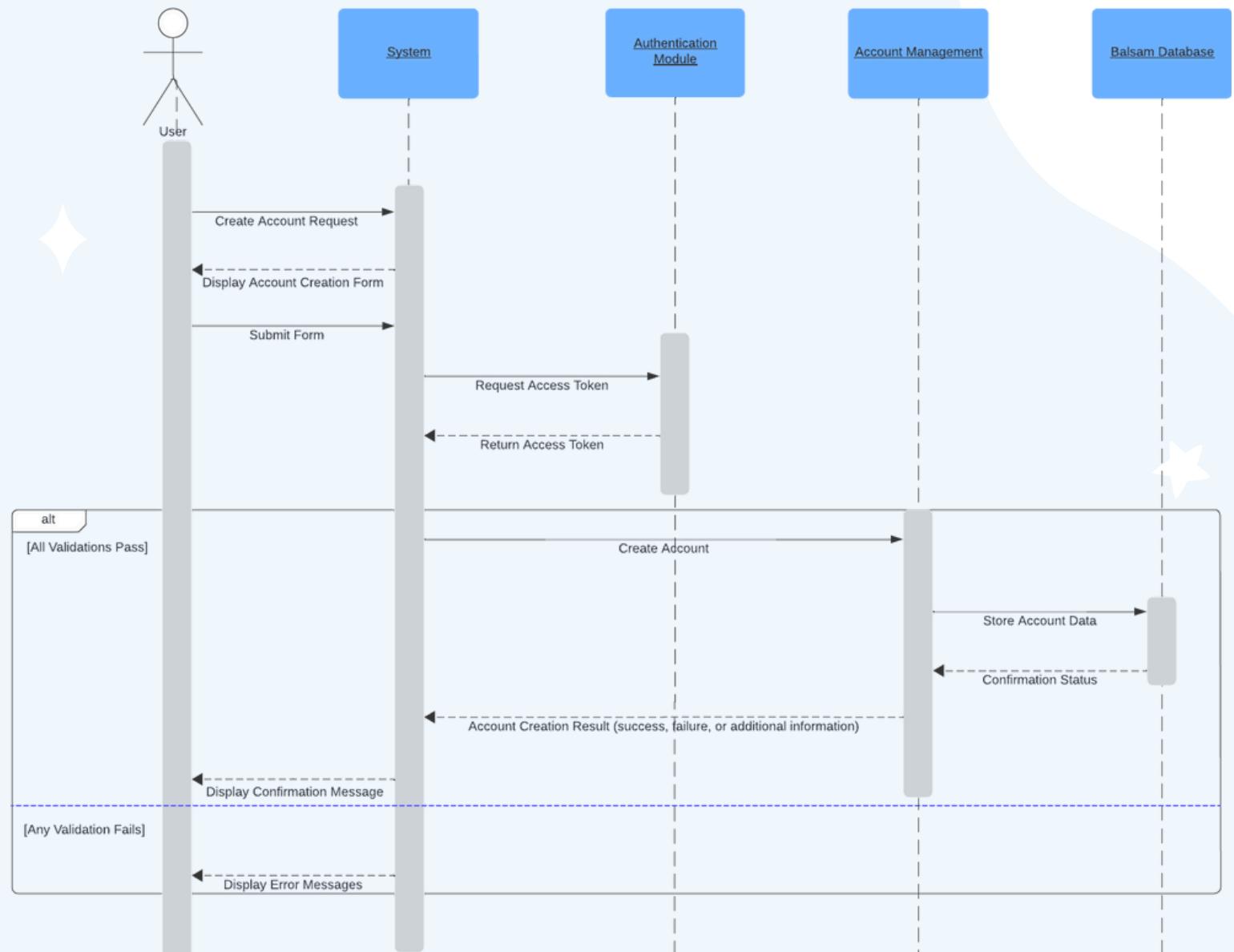
# Activity diagram for: Request delivery and installation services





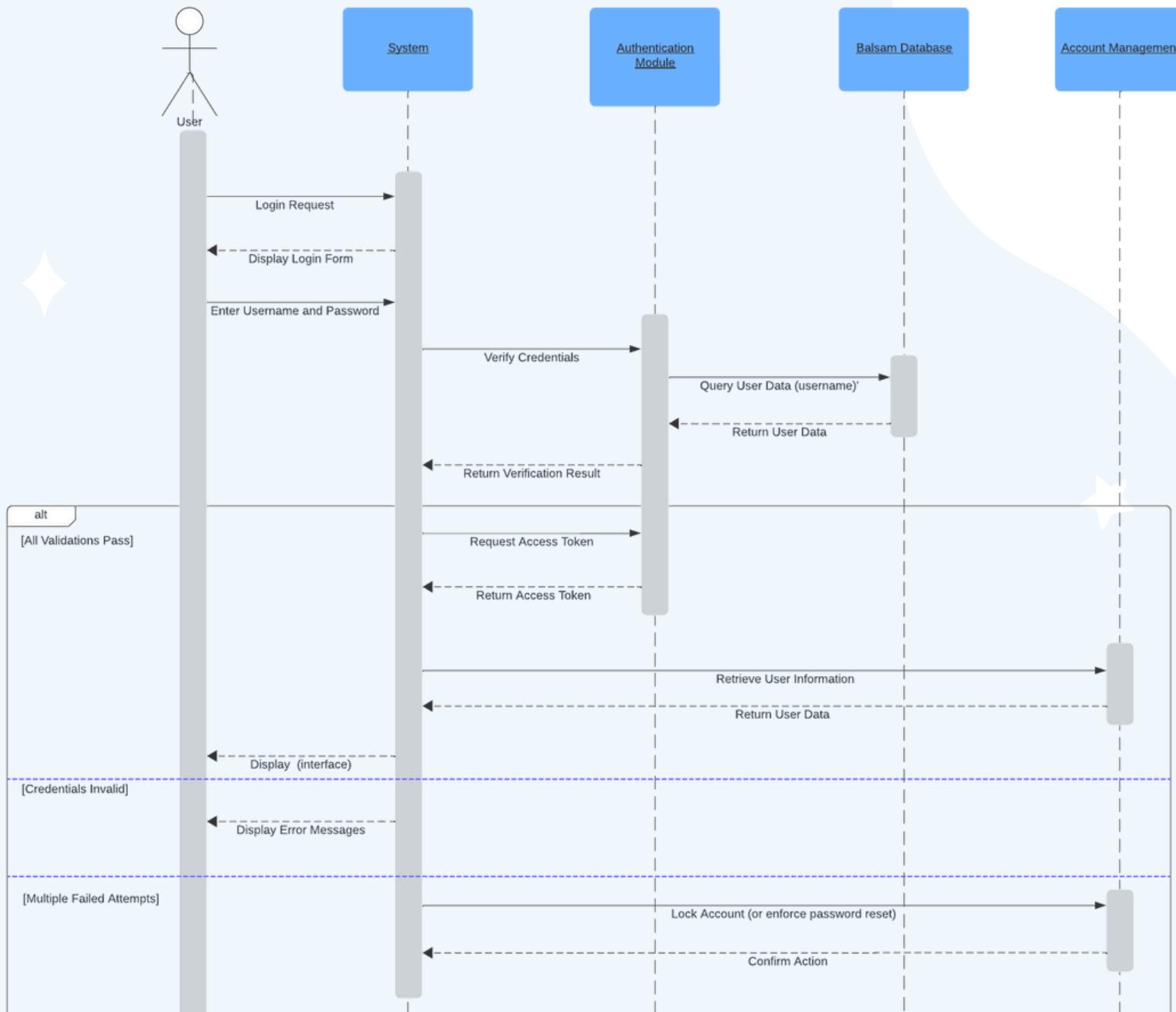
# 08 Sequence Diagrams

# Use Case: User Account Creation.



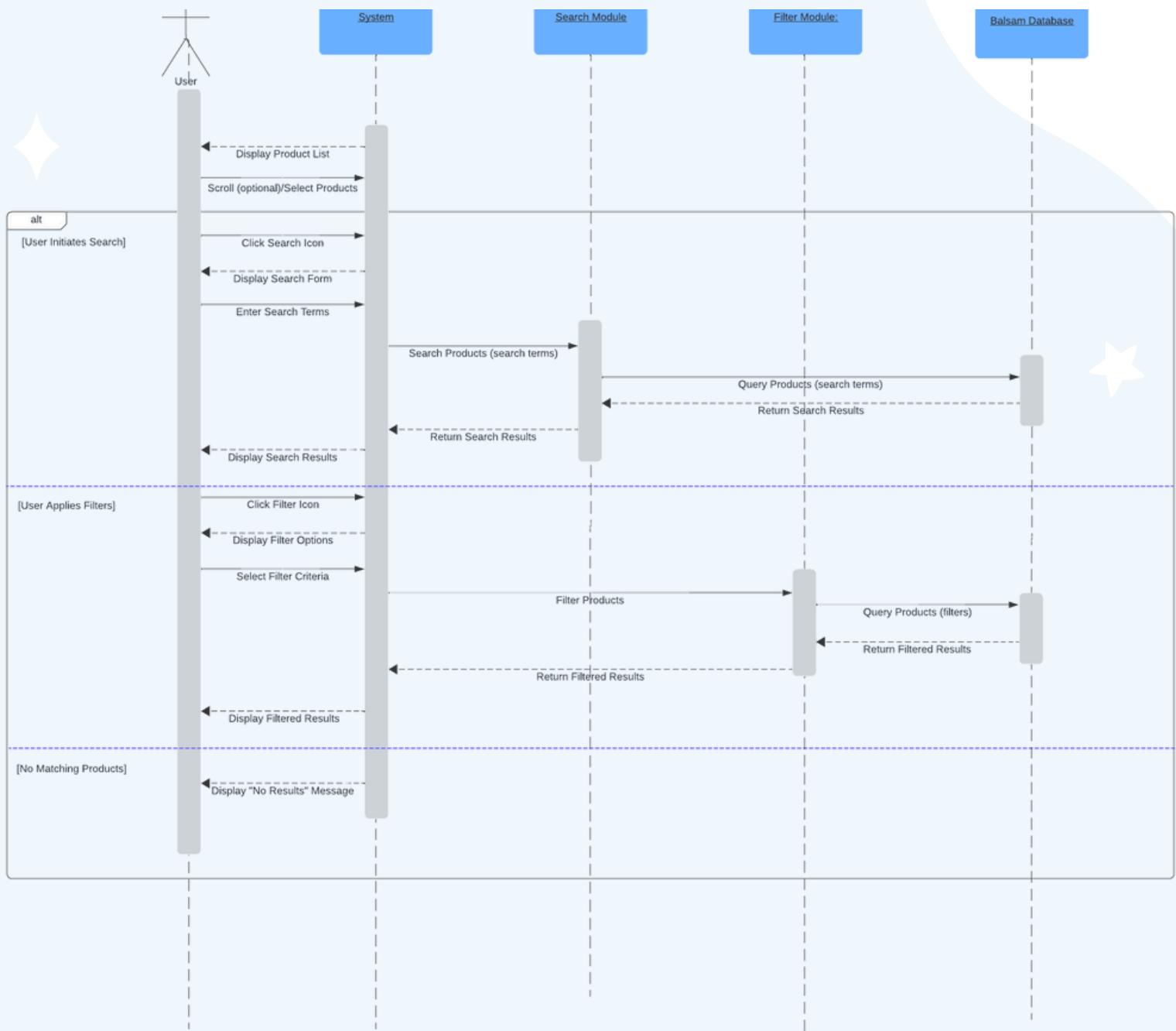
# 2. Use Case:

## User Login.



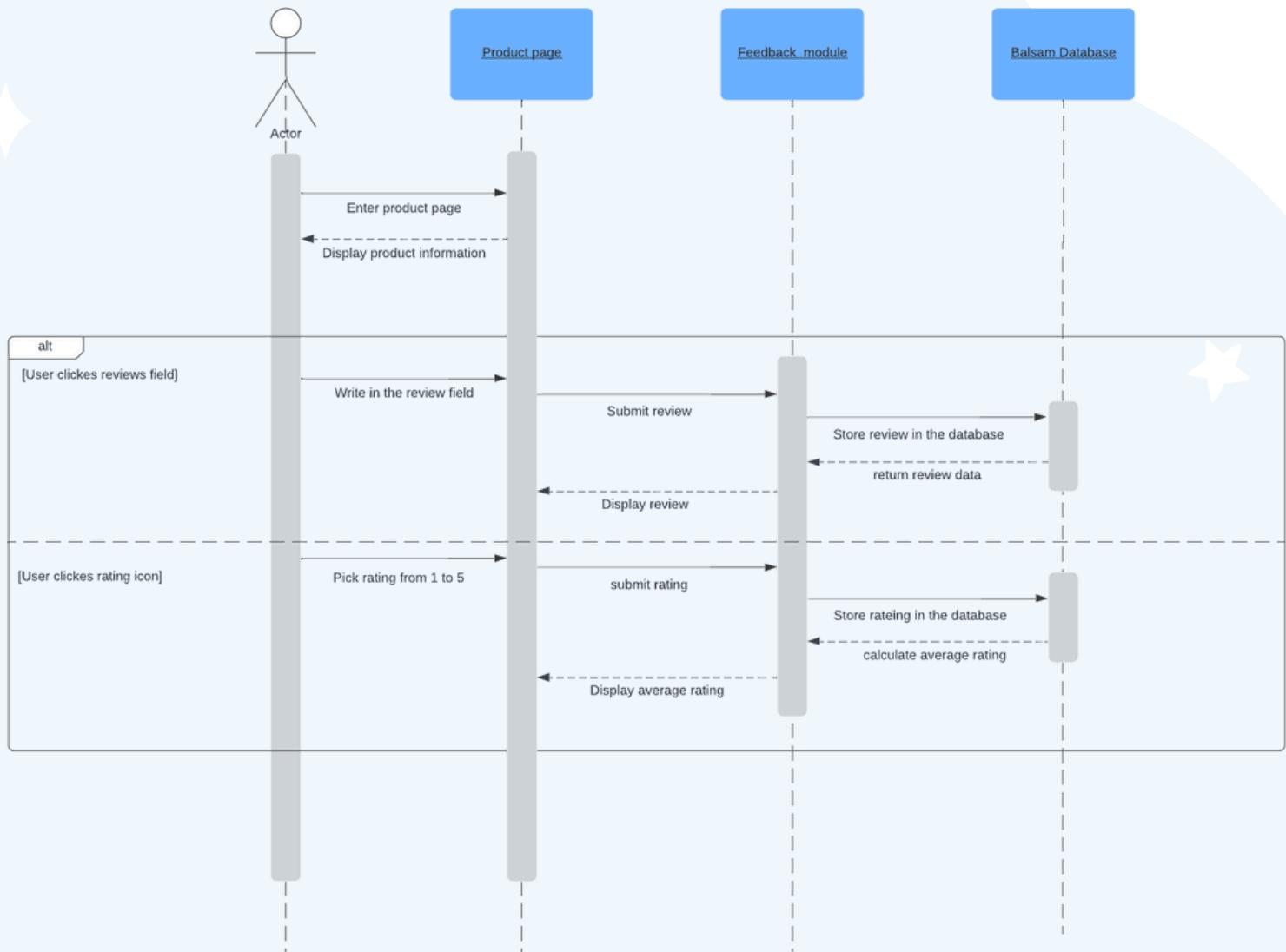
# 3. Use Case:

## Browse, Search, and Filter Products.



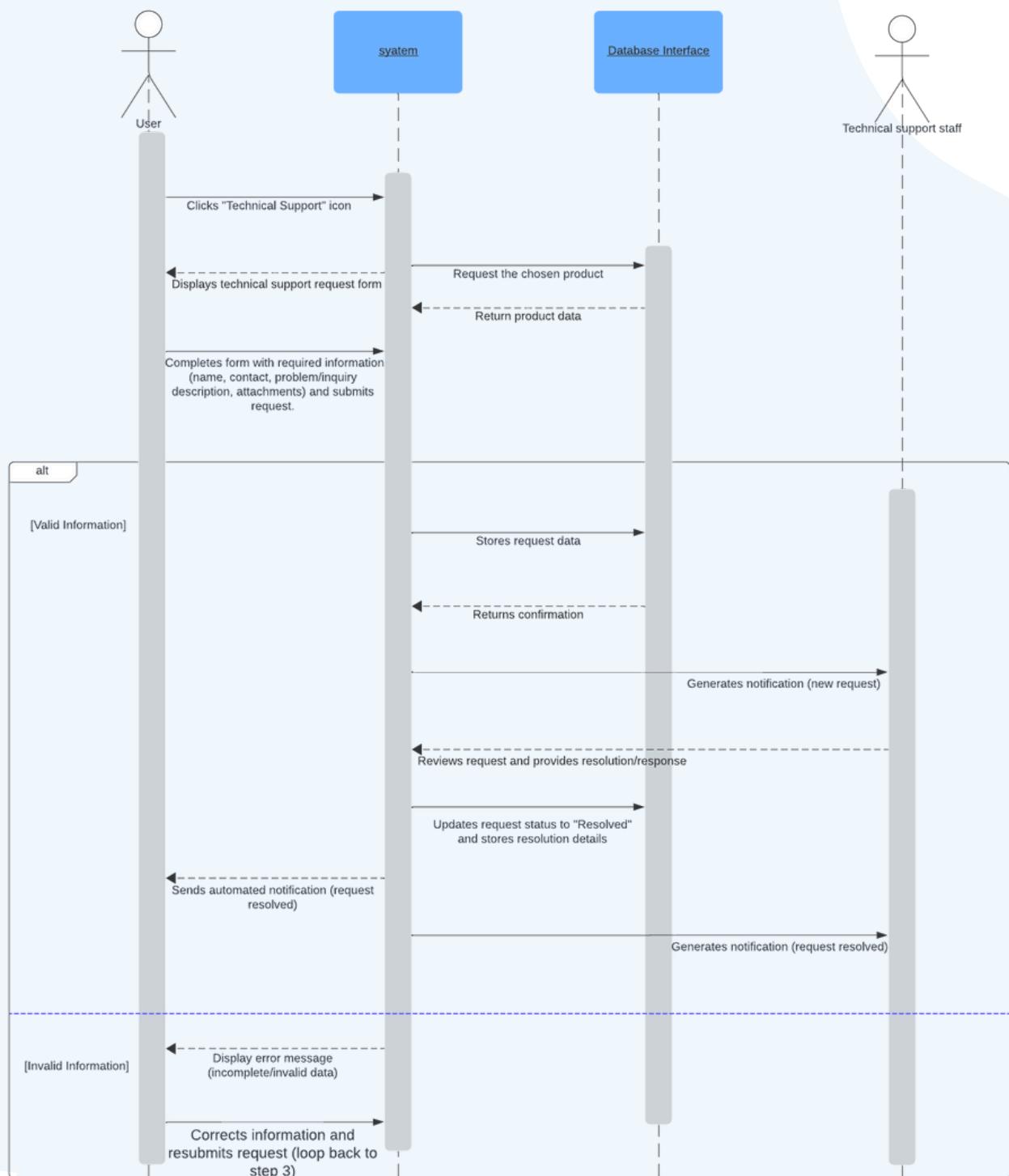
# 4 Use Case:

## Review and rate products.



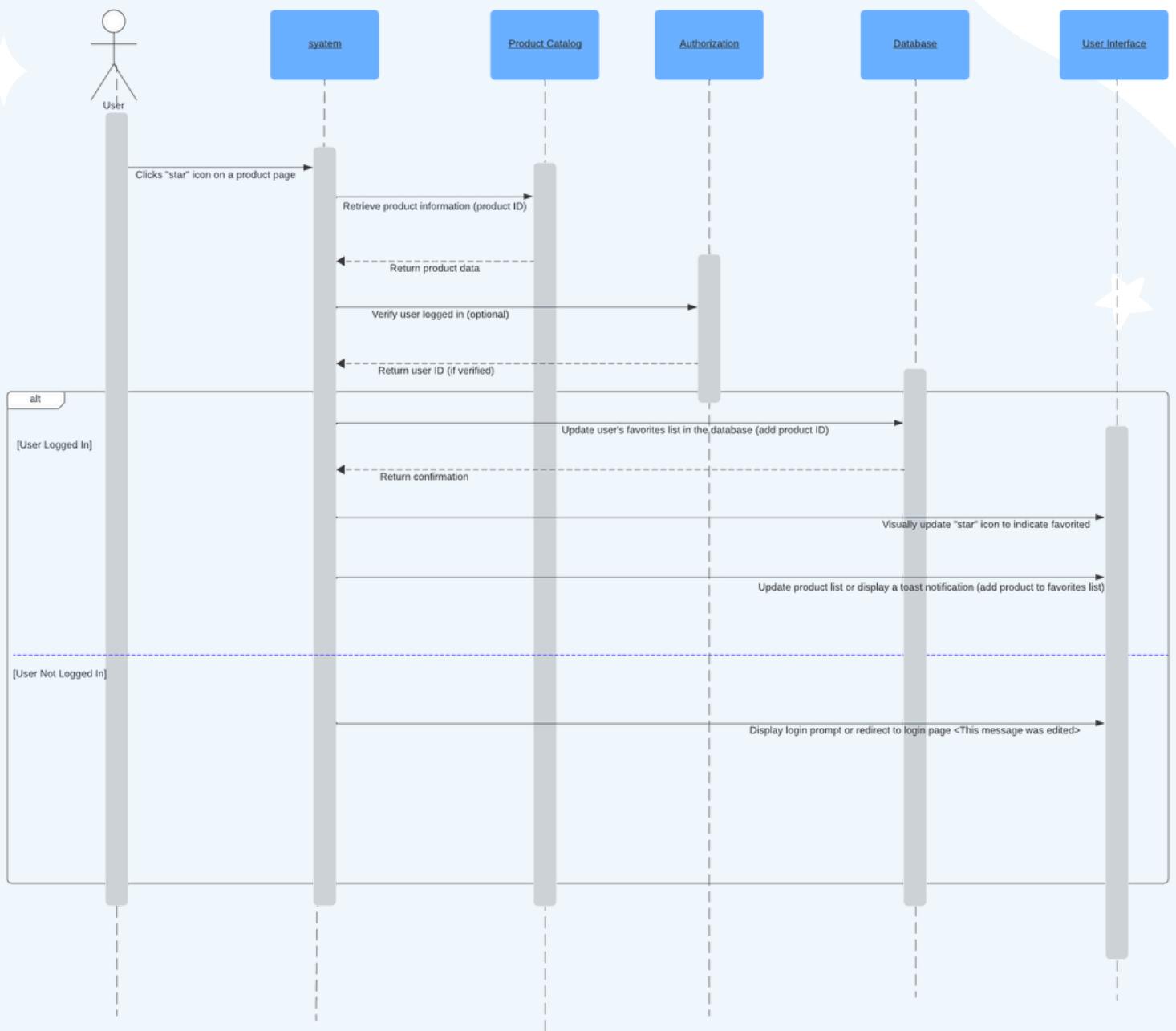
# 5. Use Case:

## provide technical support



# 6. Use Case:

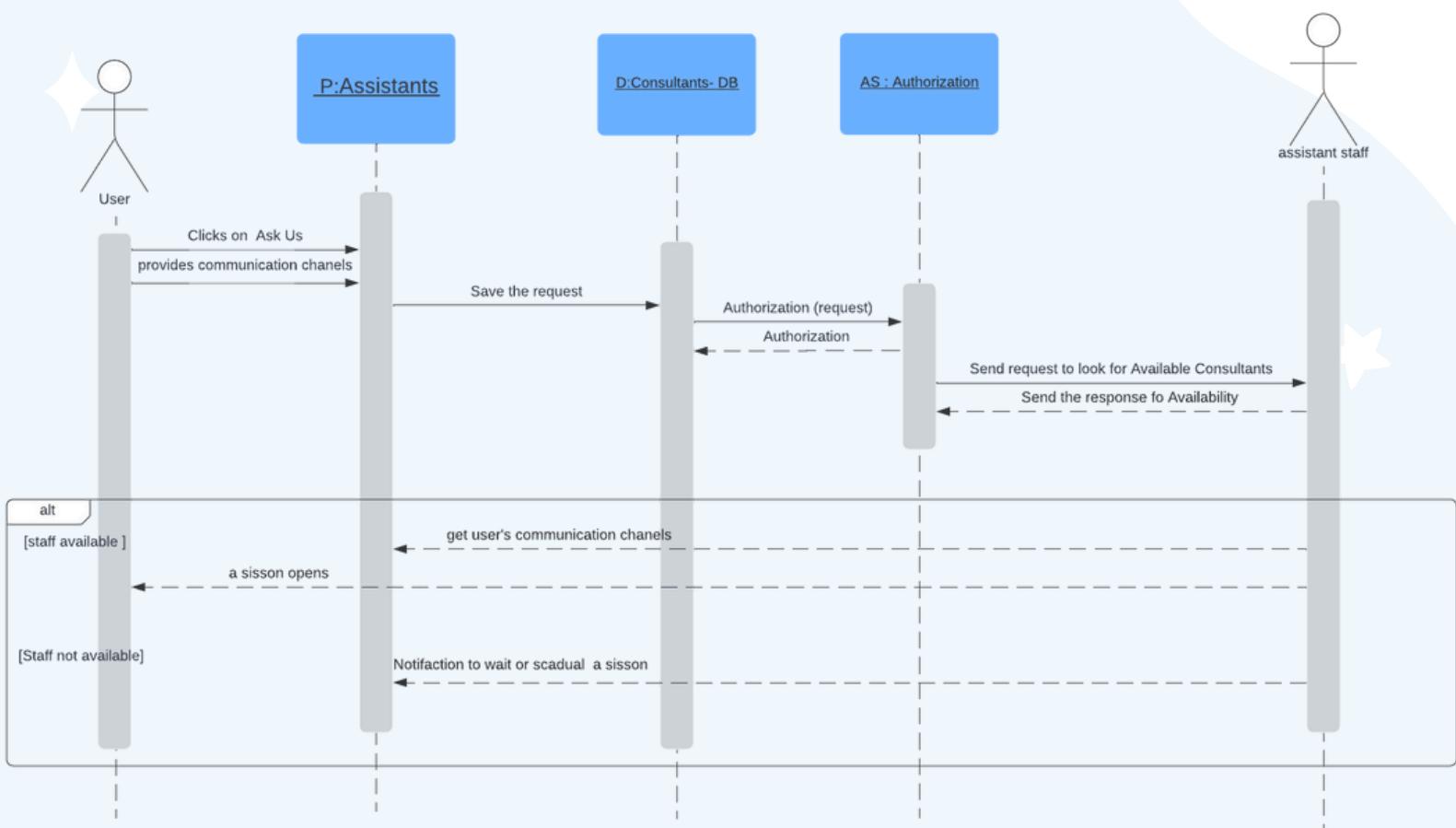
## select favorite products



# 7. Use Case:

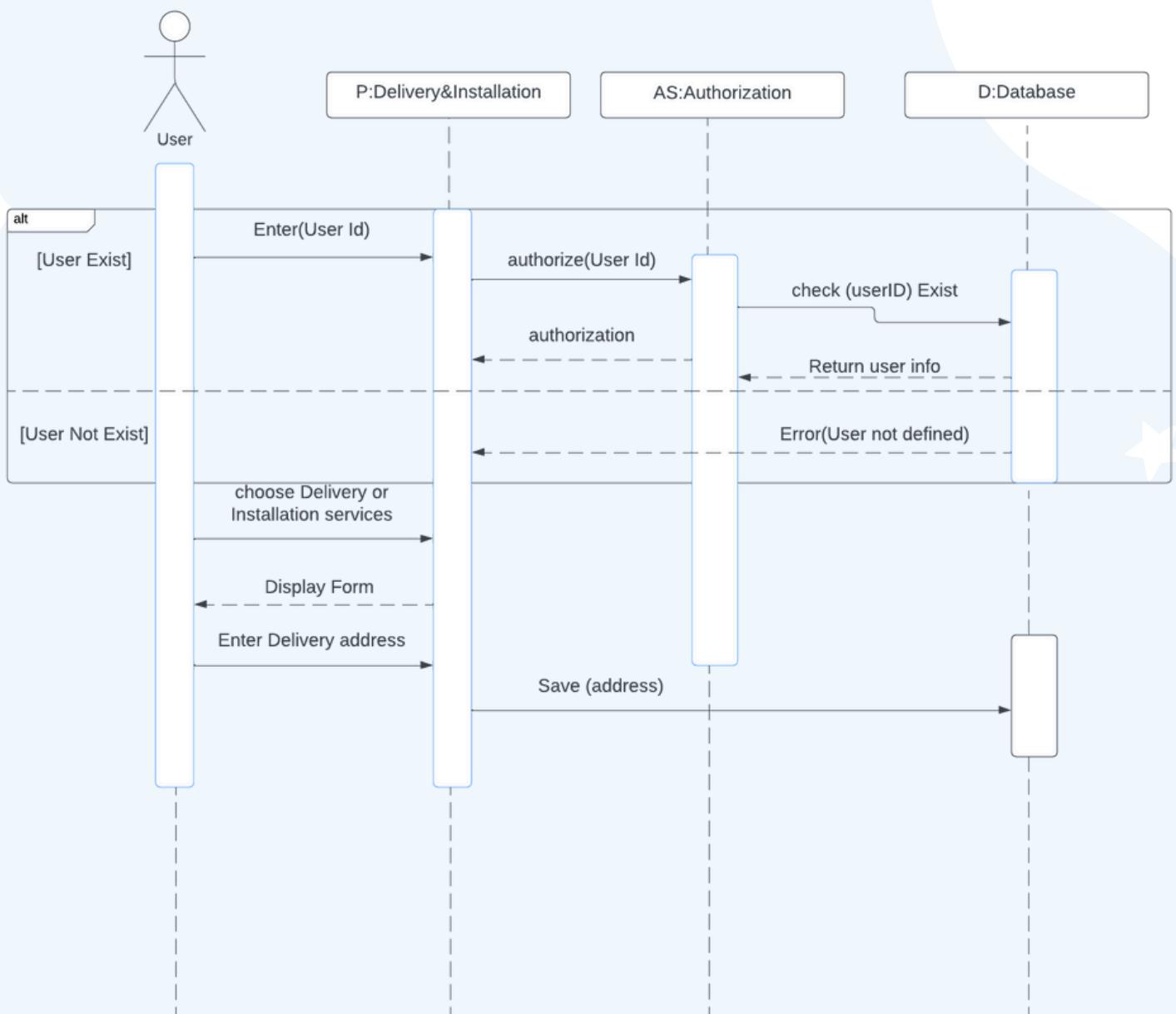


## Provide Medical Advice or Consultations.



# 8.Use Case:

## Request Delivery And Installation Services





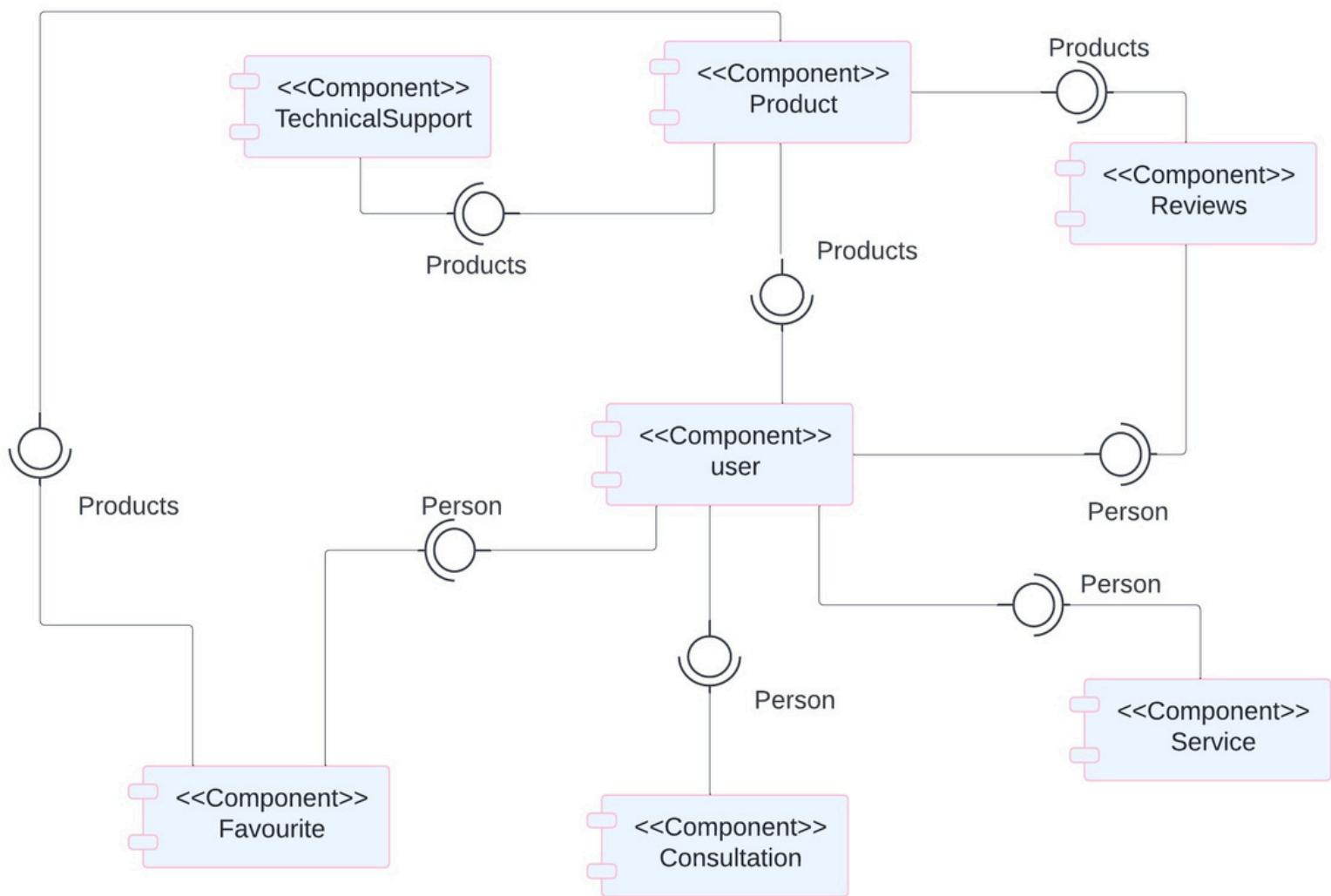
# Chapter 2

( Design phase )



# Chapter 2

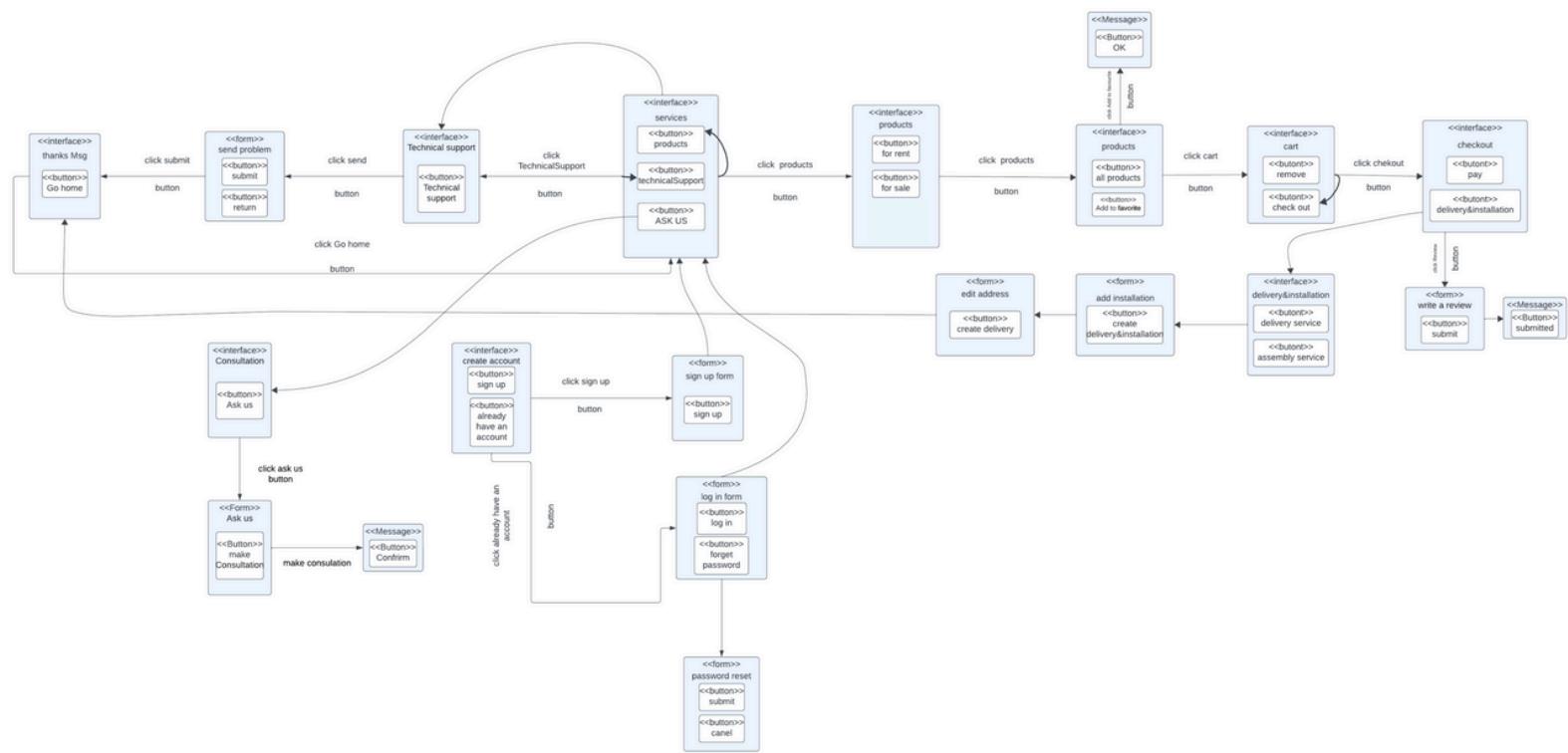
## 1. Component Design



# Chapter 2

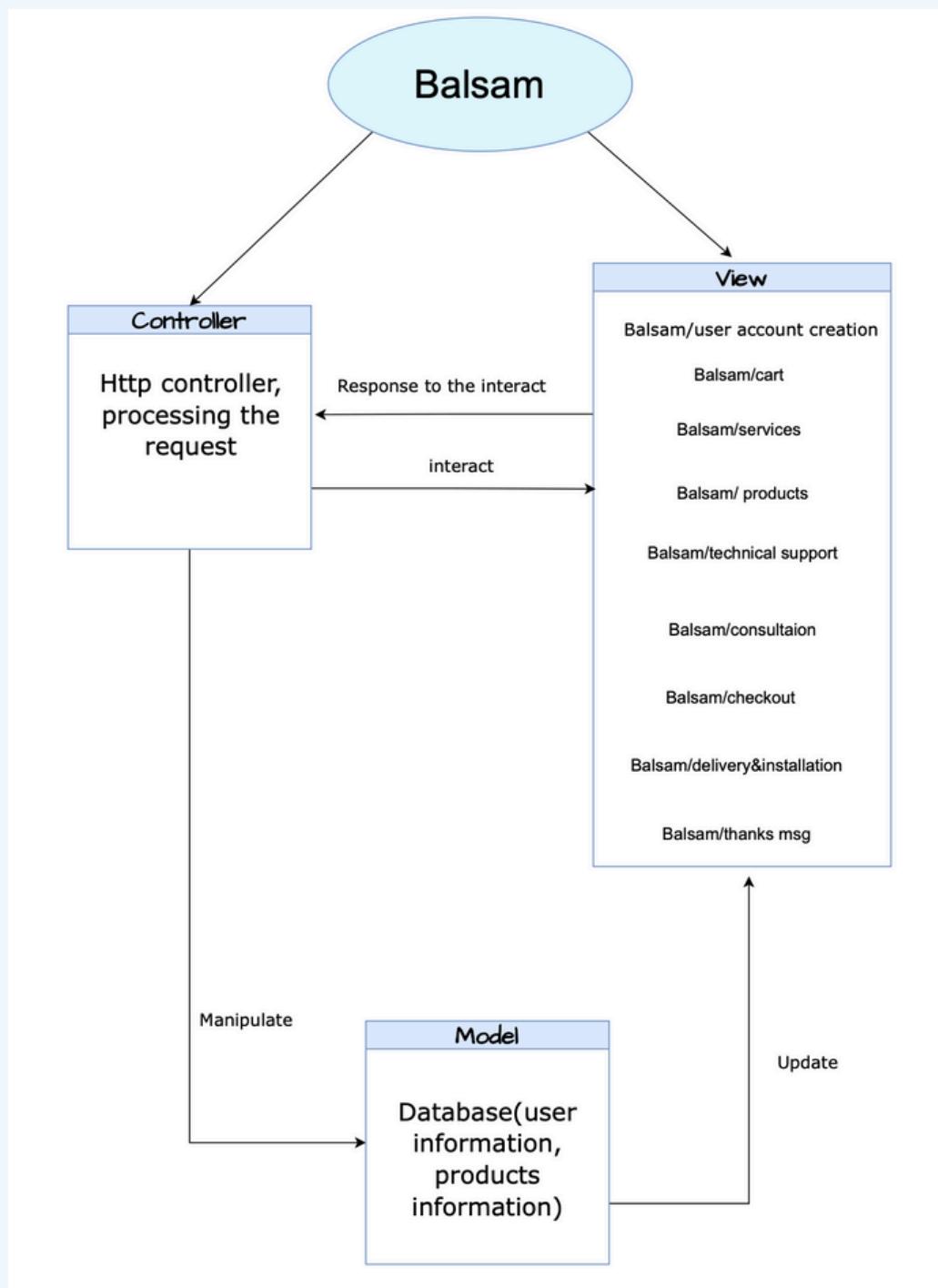


## 2. Interface Design (Windows Navigation Diagram - WND)



# Chapter 2

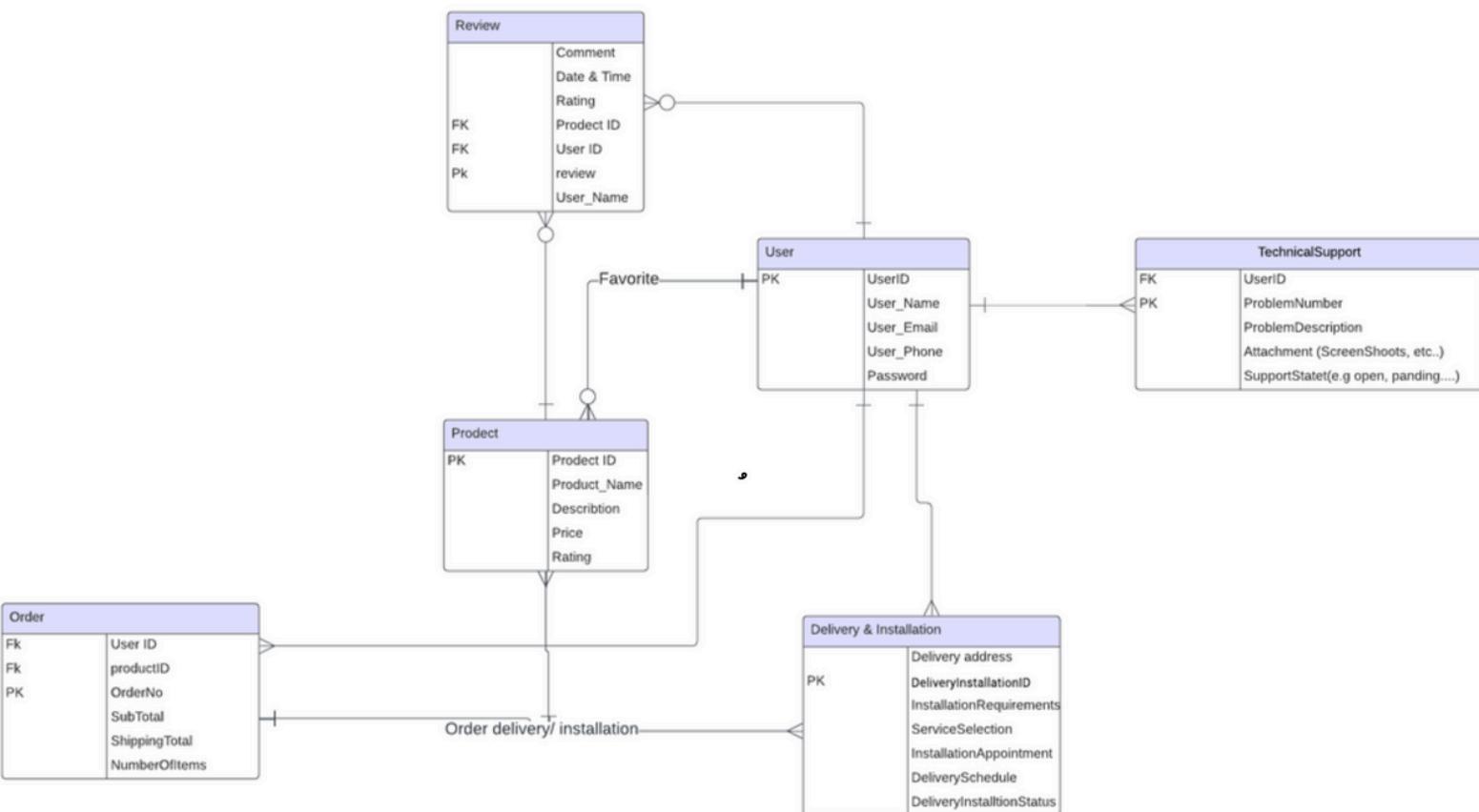
## 3. Architectural Design



# Chapter 2

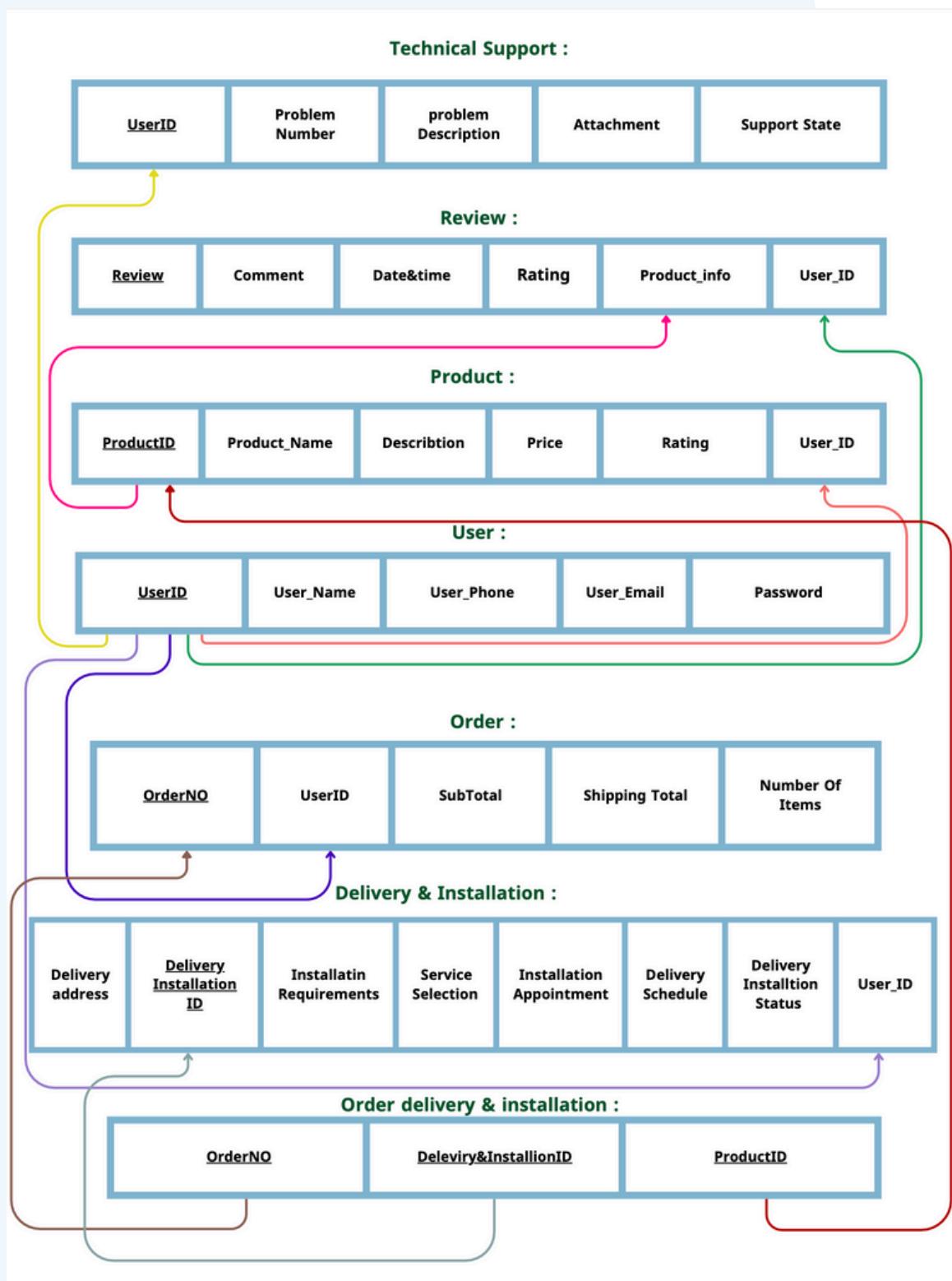


## 4. Data Storage Design ERDs



# Chapter 2

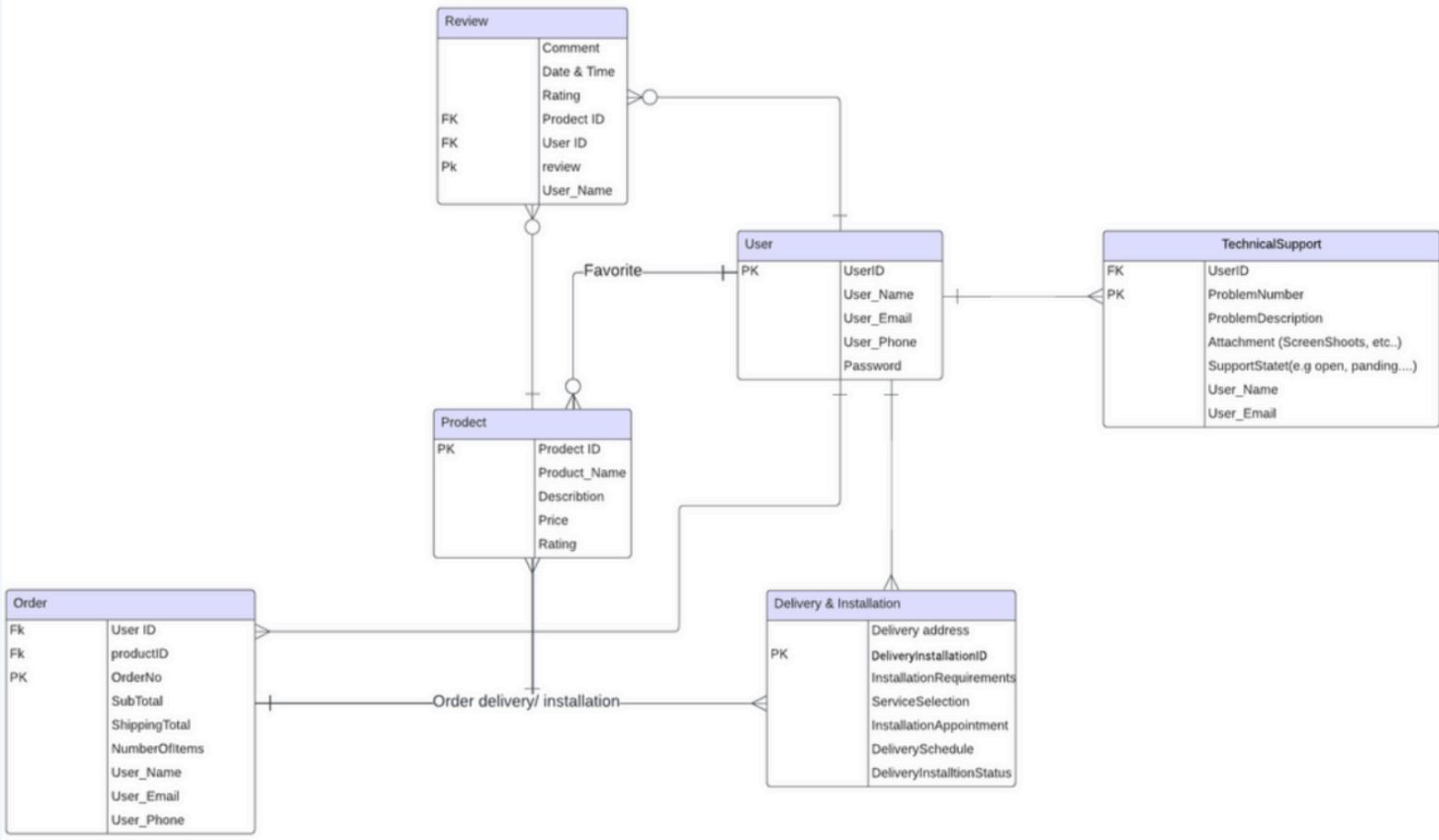
## 4. Data Storage Design Mapping into Relational Schema



# Chapter 2



## 4. Data Storage Design optimization technique: Denormalization



**Goal:**

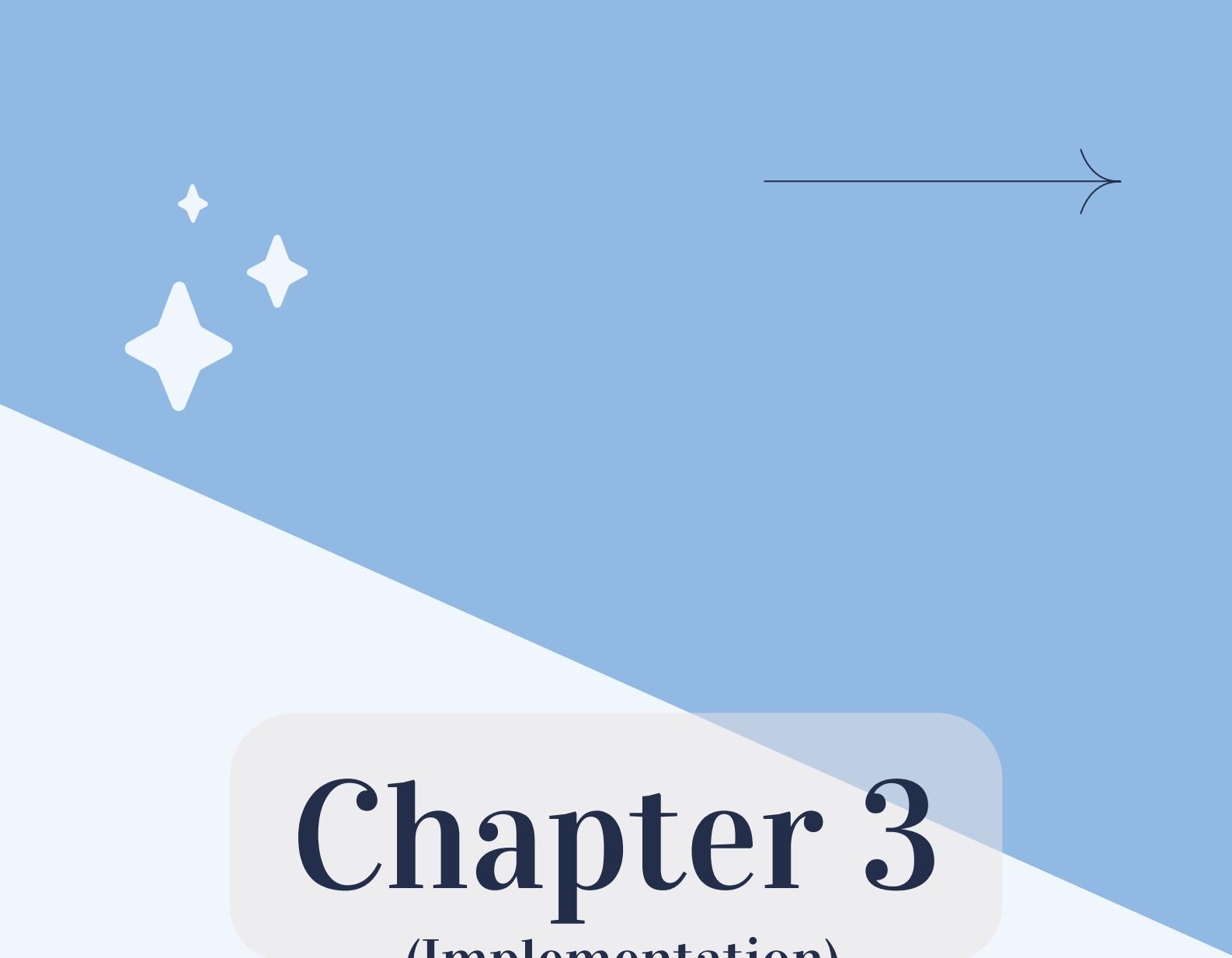
Reduce joins and simplify queries to improve performance.

**Method:** Combine related data to minimize the need for intricate joins.

**Advantages:**

1. increased readability, particularly in applications with a lot of reading.

Enhance intricate reporting and aggregations.



# Chapter 3

## (Implementation)



# Chapter 3



## Implementation :

sign up by shaima :

The image shows three side-by-side code editors, each displaying a portion of the same HTML file named "SignUp.html". The code is written in CSS and HTML, defining a sign-up form with various input fields and styling.

```
/* SignUp.html */
Users > shimherc > Desktop > SignUp.html > ...
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1.0">
6 <title>Sign Up</title>
7 <style>
8 body {
9   margin: 0;
10  padding: 0;
11  font-family: Arial, sans-serif;
12  background-color: #5382a2; /* Blue background */
13  display: flex;
14  justify-content: center;
15  align-items: center;
16  height: 100vh;
17  flex-direction: column;
18 }
19 .container {
20   width: 300px;
21   background-color: white;
22   padding: 20px;
23   border-radius: 10px;
24   box-shadow: 0 0 10px rgba(0, 0, 0, 0.5);
25 }
26 h1 {
27   text-align: center;
28   color: white;
29   margin-bottom: 20px;
30 }
31 h2 {
32   text-align: center;
33   color: #5382a2;
34 }
35 input[type="text"],
36 input[type="password"],
37 input[type="email"],
```

```
/* SignUp.html */
Users > shimherc > Desktop > SignUp.html > ...
1 <!DOCTYPE html lang="en">
2 <html lang="en">
3 <head>
4 <style>
5   input[type="text"],
6   input[type="password"],
7   input[type="email"],
8   input[type="tel"],
9   input[type="submit"] {
10    width: 100%;
11    padding: 10px;
12    margin: 8px 0;
13    display: inline-block;
14    border: 1px solid #ccc;
15    border-radius: 4px;
16    box-sizing: border-box;
17  }
18  input[type="submit"] {
19    background-color: #a9cce9;
20    color: white;
21    cursor: pointer;
22  }
23  input[type="submit"]:hover {
24    background-color: #0056b3;
25  }
26  .btns {
27    text-align: center;
28  }
29  .btns a {
30    color: #222222;
31    text-decoration: none;
32    margin-right: 10px;
33    font-size: 10px;
34    font-family: Lucida Console;
35    font-weight: bold;
36  }
37  .btns a:hover {
38    text-decoration: underline;
39 }
```

```
/* SignUp.html */
Users > shimherc > Desktop > SignUp.html > ...
1 <html lang="en">
2 <head>
3 <style>
4   label {
5     font-size: 12px;
6     color: #333;
7     display: block;
8     margin-bottom: 5px;
9   }
10  </style>
11  </head>
12  <body>
13  <h1>Balsam(بالسام)</h1>
14  <div class="container">
15    <h2>Sign Up</h2>
16    <form action="#" method="post">
17      <label for="fullname">Full Name:</label>
18      <input type="text" placeholder="Enter Full Name" name="fullname" required>
19      <label for="email">Email:</label>
20      <input type="email" placeholder="Enter Email Address" name="email" required>
21      <label for="username">User ID:</label>
22      <input type="text" placeholder="Enter User ID" name="username" required>
23      <label for="password">Password:</label>
24      <input type="password" placeholder="Enter Password" name="password" required>
25      <label for="phone">Phone Number:</label>
26      <input type="tel" placeholder="Enter Phone Number" name="phone" required>
27      <input type="submit" value="Sign Up">
28    </form>
29    <div class="btns">
30      <a href="balsam.html">already have an account?<br></a>
31    </div>
32  </div>
33  </body>
34  </html>
```



# Chapter 3



## Implementation :

### log in by Khadeejh :

The image shows a code editor with two tabs open. The top tab is 'login\_baham.html' and the bottom tab is 'login\_baham.css'. Both tabs have the same line count of 100 lines.

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Login </title>
<style>
body {
    margin: 0;
    padding: 0;
    font-family: Arial, sans-serif;
    background-color: #5382a2; /* Blue background */
    display: flex;
    justify-content: center;
    align-items: center;
    height: 100vh;
    flex-direction: column;
}
.container {
    width: 300px;
    background-color: white;
    padding: 20px;
    border-radius: 10px;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.5);
}
h1 {
    text-align: center;
    color: white;
    margin-bottom: 20px;
}
h2 {
    text-align: center;
    color: #5382a2;
}
input[type="text"],
input[type="password"],
input[type="submit"] {
    width: 100px;
    padding: 10px;
    margin: 10px 0;
    display: inline-block;
    border: 1px solid #ccc;
    border-radius: 4px;
    box-sizing: border-box;
}
input[type="submit"] {
    background-color: #a9cce9;
    color: white;
    cursor: pointer;
}
input[type="submit"]:hover {
    background-color: #0056b3;
}
.btns {
}
.btns a {
    color: #222222;
    text-decoration: none;
    margin-right: 10px;
    font-size: 10px;
    font-family: Lucida Console ;
    font-weight: bold;
}
.btns a:hover {
    text-decoration: underline;
}
label {
    font-size: 12px;
    color: #333;
    display: block;
    margin-bottom: 5px;
}
</style>
</head>
<body>
<h1>Balsam(بسم)</h1>
<div class="container">
<h2>Log In</h2>
<form action="#" method="post">
<label form="username">Username:</label>
<input type="text" placeholder="Enter Username...." name="username" required>
<label form="password">Password:</label>
<input type="password" placeholder="Enter Password...." name="password" required>
<input type="submit" value="Log In">
</form>
<div class="btns">
    <a href="#">FORGET PASSWORD ?<br></a>
    <a href="#">SIGN UP </a>
</div>
</div>
</body>
</html>
```



# Chapter 3



## Implementation :

### services by jana:

#### Remarks

The screenshot shows two Sublime Text windows side-by-side. The left window, titled 'Home.html', displays the HTML code for a website. The right window, titled 'styles.css', displays the corresponding CSS styles.

```
Sublime Text File Edit Selection Find View Goto Tools Project Window Help
Home.html
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Home Page</title>
7   <link rel="stylesheet" href="styles.css"> <!-- Link to your external CSS file -->
8 </head>
9 <body>
10  <header class="toolbar">
11    <!-- Logo for the toolbar -->
12    <div class="logo">
13      <a href="login_balsam.html">
14        
15      </a>
16    </div>
17  </header>
18
19 <main class="content">
20   <div class="image-grid">
21     <!-- Each image links to a different page -->
22     <div class="image-item">
23       <a href="page1.html">
24         
25         <h3>Products</h3>
26       </a>
27     </div>
28
29     <div class="image-item">
30       <a href="installation.html">
31         
32         <h3>Installation</h3>
33       </a>
34     </div>
35
36     <div class="image-item">
37       <a href="page3.html">
38         
39         <h3>Technical Support</h3>
40       </a>
41     </div>
42
43     <div class="image-item">
44       <a href="Consultation.html">
45         
46         <h3>ASK US!</h3>
47       </a>
48     </div>
49   </div>
50
51 </body>
52 </html>
```

```
styles.css
1 /* Reset default margin and padding */
2 body, html {
3   margin: 0;
4   padding: 0;
5 }
6
7 /* Basic styling */
8 .toolbar {
9   background-color: #ffffff; /* White background for the toolbar */
10  padding: 10px 0;
11  display: flex;
12  justify-content: space-between; /* Align logo and nav links to opposite ends */
13  align-items: center; /* Center vertically */
14  border-bottom: 2px solid #dcdcdc; /* Gray border (underside) */
15 }
16
17 .toolbar .logo img {
18   height: 80px; /* Adjust logo height as needed */
19   padding-left: 20px; /* Add padding to separate logo from nav links */
20 }
21
22 .toolbar .nav {
23   text-align: center;
24 }
25
26 .toolbar .nav ul {
27   list-style: none;
28   padding: 0;
29 }
30
31 .toolbar .nav ul li {
32   display: inline;
33   margin-right: 20px;
34 }
35
36 .toolbar .nav ul li a {
37   color: #5382a2; /* Dark blue color for nav links */
38   text-decoration: none;
39   font-weight: bold;
40 }
41
42 .content {
43   max-width: 1200px;
44   margin: 0 auto;
45   padding: 20px;
46 }
47
48 .image-grid {
49   display: flex;
50   justify-content: space-around;
51   flex-wrap: wrap;
52 }
53
54 .image-item {
55   text-align: center;
56   margin-bottom: 20px;
57   width: 45%; /* Set width to fit two items in a row */
58 }
```

```
.image-item a {
  text-decoration: none;
  color: #5382a2;
}

.image-item img {
  width: 50%; /* Make images responsive within their container */
  border-radius: 10px;
  object-fit: cover;
}

.image-item h3 {
  margin-top: 10px;
  font-size: 18px;
}
```



# Chapter 3



## Implementation :

### consultation by Dhay:

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Consultation</title>
<style>
body {
    margin: 0;
    padding: 0;
    font-family: Arial, sans-serif;
    background-color: #f5f5f5; /* Light gray background */
}
.toolbar {
    background-color: #5382a2;
    color: white;
    padding: 10px 20px;
    display: flex;
    justify-content: center; /* Center the text horizontally */
    align-items: center;
}
.toolbar h1 {
    margin: 0;
}
.container {
    margin: 100px auto;
    width: 80%;
    max-width: 600px;
    background-color: white;
    padding: 20px;
    border-radius: 10px;
    box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);
}
h2 {
    text-align: center;
    color: #5382a2;
    display: none; /* Hide the heading */
}
label {
    font-size: 16px;
    color: #333;
    display: block;
    margin-bottom: 8px;
}
input[type="text"],
textarea {
    width: 100%;
    padding: 10px;
    margin-bottom: 15px;
    border: 1px solid #ccc;
    border-radius: 4px;
    box-sizing: border-box;
}
input[type="text"] {
    color: #333;
    display: block;
    margin-bottom: 8px;
}
input[type="text"],
textarea {
    width: 100%;
    padding: 10px;
    margin-bottom: 15px;
    border: 1px solid #ccc;
    border-radius: 4px;
    box-sizing: border-box;
}
textarea {
    resize: vertical;
    min-height: 100px;
}
input[type="submit"] {
    background-color: #5382a2;
    color: white;
    padding: 12px 20px;
    border: none;
    border-radius: 4px;
    cursor: pointer;
    font-size: 16px;
}
input[type="submit"]:hover {
    background-color: #0056b3;
}
</style>
</head>
<body>
<div class="toolbar">
    <h1>Consultation Form</h1>
</div>
<div class="container">
    <form action="#" method="post">
        <label for="email">Your Email:</label>
        <input type="text" id="email" name="email" required>
        <label for="consultation">Brief Description of Consultation:</label>
        <textarea id="consultation" name="consultation" required></textarea>
        <input type="submit" value="Create a Consultation">
    </form>
</div>
</body>
</html>
```

# Chapter 3

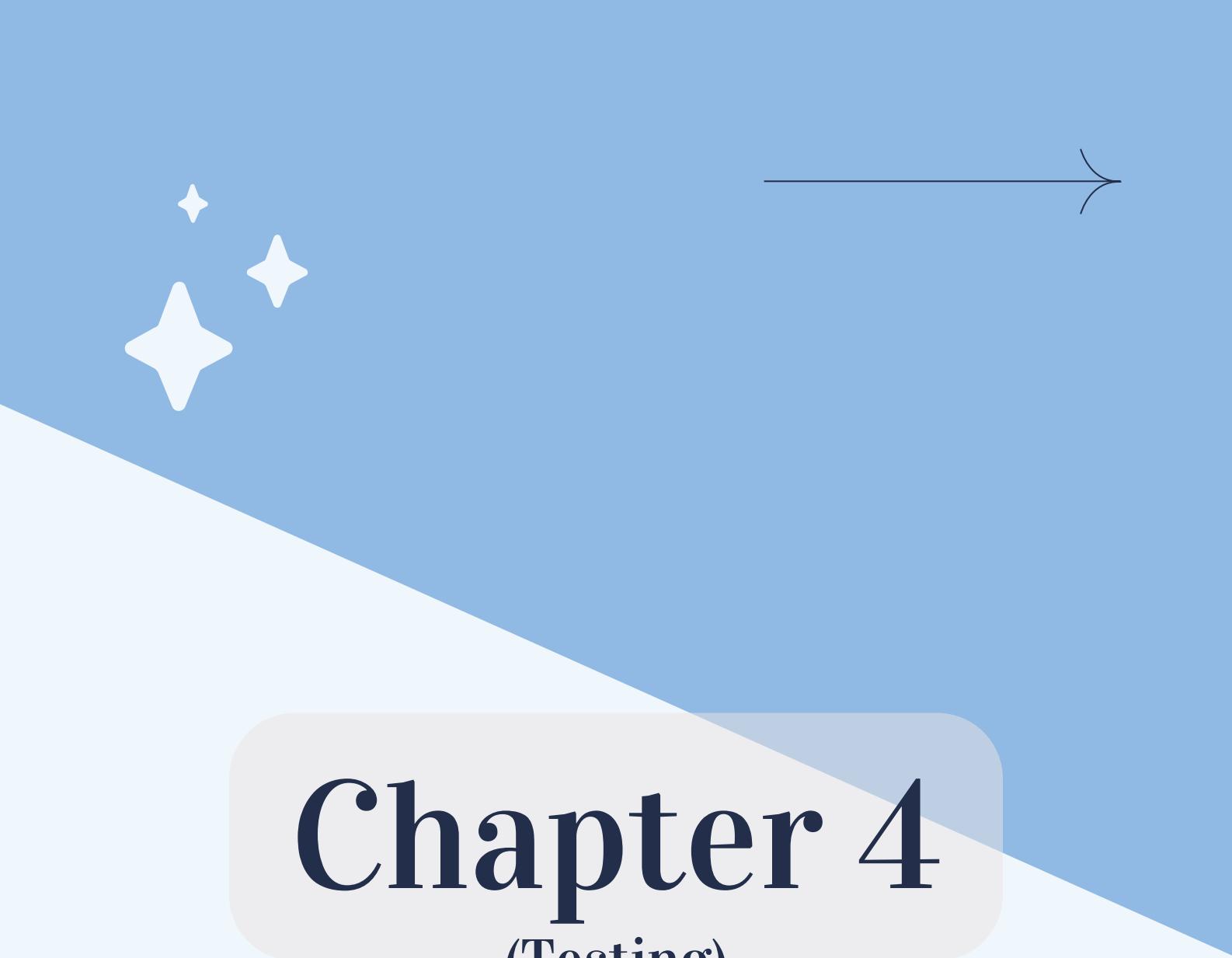


## Implementation :

### installation by Ghadah:

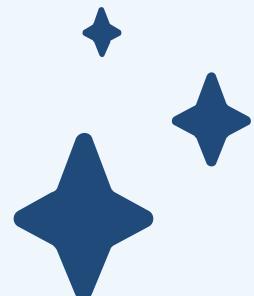
```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  <link rel="stylesheet" href="style.css">
5  <meta charset="UTF-8">
6  <meta name="viewport" content="width=device-width, initial-scale=1.0">
7  <title>Add installation</title>
8  <link rel="stylesheet" href="style.css">
9  </head>
10 <body>
11
12 <div class="topnav">
13   <a class="active" href="#home"><</a>
14   <a href="#">Add Installation</a>
15   <a href="#">About</a>
16 </div>
17   <h2>Add Installation</h2>
18 <div class="container">
19   <form action="#" method="post">
20     <label>Installation device</label>
21     <input type="text" placeholder="Enter installation device" name="installationDevice" required>
22     <label>Date & Time:</label>
23     <input type="date" placeholder="Enter Email">
24     <input type="submit" value="CREATE INSTALLATION">
25   </form>
26   <div class="button">
27     <a href="#"></a>
28   </div>
29 </div>
30 </body>
31 </html>
```

```
body {
  padding: 0;
  font-family: Arial, sans-serif;
  background-color: #f6f6f6;
  justify-content: center;
  align-items: center;
  height: 100vh;
  flex-direction: column;
}
h1 {
  text-align: center;
  color: white;
  margin-bottom: 20px;
}
h2 {
  text-align: center;
  margin-bottom: 70px;
  color: black;
}
input[type="text"],
input[type="password"],
input[type="date"],
input[type="submit"] {
  width: 70%;
  height: 70px;
  padding: 10px;
  margin: 8px 10px 8px 10px;
  display: block;
  border: 0.2px solid #ccc;
  border-radius: 0px;
  box-sizing: border-box;
  margin-left: 50px;
  justify-content: left;
  align-items: left;
  display: flex;
}
input[type="submit"] {
  background-color: #a9cce9;
  margin-top: 30px;
  color: white;
  justify-content: center;
  display: flex;
  cursor: pointer;
}
input[type="submit"]:hover {
  background-color: #0056b3;
  color: white;
}
button {
  text-align: center;
}
button a {
  color: #222222;
  text-decoration: none;
  margin-right: 10px;
  font-size: 10px;
  font-family: Lucida Console;
  font-weight: bold;
}
button a:hover {
  text-decoration: underline;
}
.container{
  margin-left: 220px;
}
label {
  font-size: 12px;
  color: #333;
  display: block;
  margin-bottom: 5px;
  margin-top: 10px;
}
.topnav {
  background-color: #a9cce9;
  margin-bottom: 80px;
  padding: 8px;
}
.topnav a {
  color: #f2f2f2;
  text-align: top;
  padding: 20px 20px 20px 60px;
  text-decoration: none;
  margin-left: 185px;
  font-size: 17px;
}
.topnav a:hover {
  background-color: #ddd;
  color: black;
}
```



# Chapter 4

(Testing)



# Chapter 4



## Testing

Test Case	Input	Expected output	Actual output	Remarks	By:
Empty field in create a consultation	Empty consultation information	An error message will appear	An error message appears	Pass	Dhay
Create a delviery	Delviery information	it will add the delivery information to the database and display a page with a confirmation message	added the delivery information in DataBase and display a page with a confirmation message	Pass	Dhay
Access to all services	Click on each button	Go to the service screen	Successfully navigated to the expected screen	Pass	Jana
Empty entering technical support form	Empty input	Empty input detected	An error message appears	Pass	Jana



# Chapter 4



## Testing

Test Case	Input	Expected output	Actual output	Remarks	By:
access the product for sale screen	click on -for sale- button	go to the for sale screen	successfully navigated to the expected screen	Pass	Ghadah
access the product detail screen	click on view button of the specific product	go to the specified product detail screen	successfully navigated to the expected screen	Pass	Ghadah
Empty field in create account	Empty user information	error message will be in screen	error message	Pass	Khadeejh
Reset password	User Email	After Clicking on button submit and it will send a message to email to reset password	Successfully send a message, a temporary password	Pass	Khadeejh



# Chapter 4



## Testing

Test Case	Input	Expected output	Actual output	Remarks	By:
Log In	Email Password	Move to the services screen	Successfully navigated to the expected screen	Pass	Shaima
Password reset by wrong email	Email	send a error message	error message	Pass	Shaima

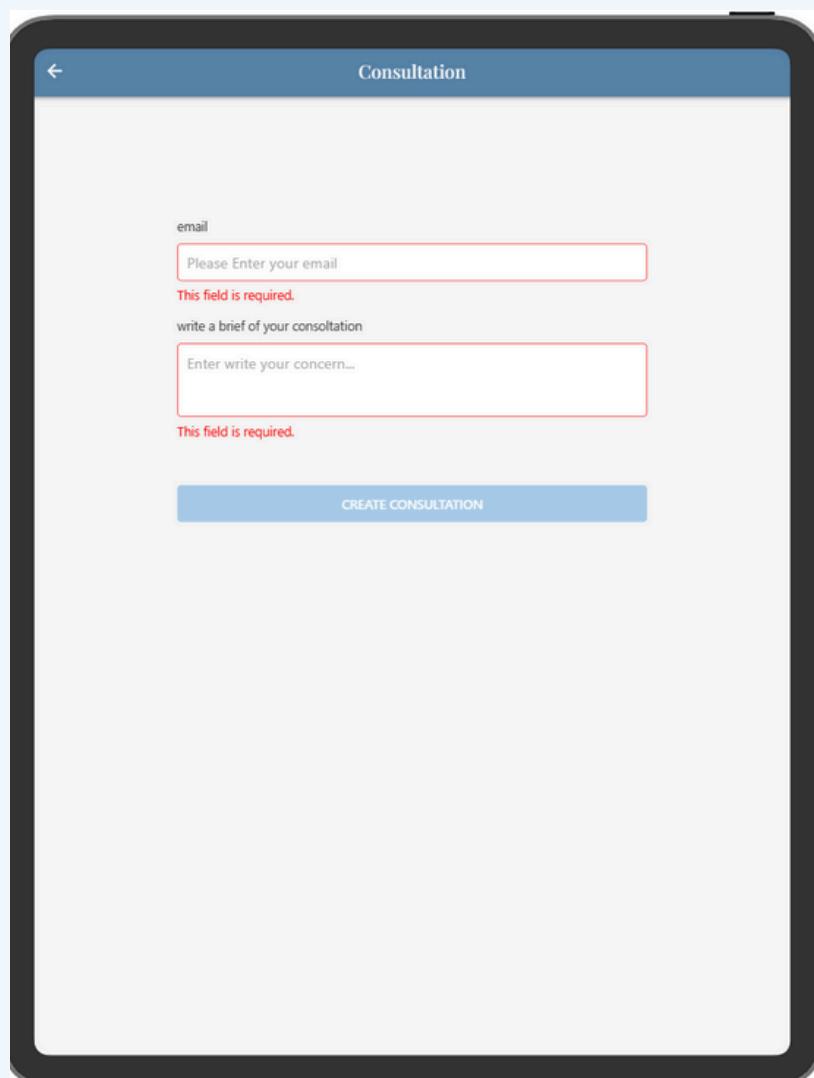
# Chapter 4



## Testing by Dhay

### 1. Empty field in create a consultation

**Description:** Ensure that an error message is displayed to the user when the field is left empty



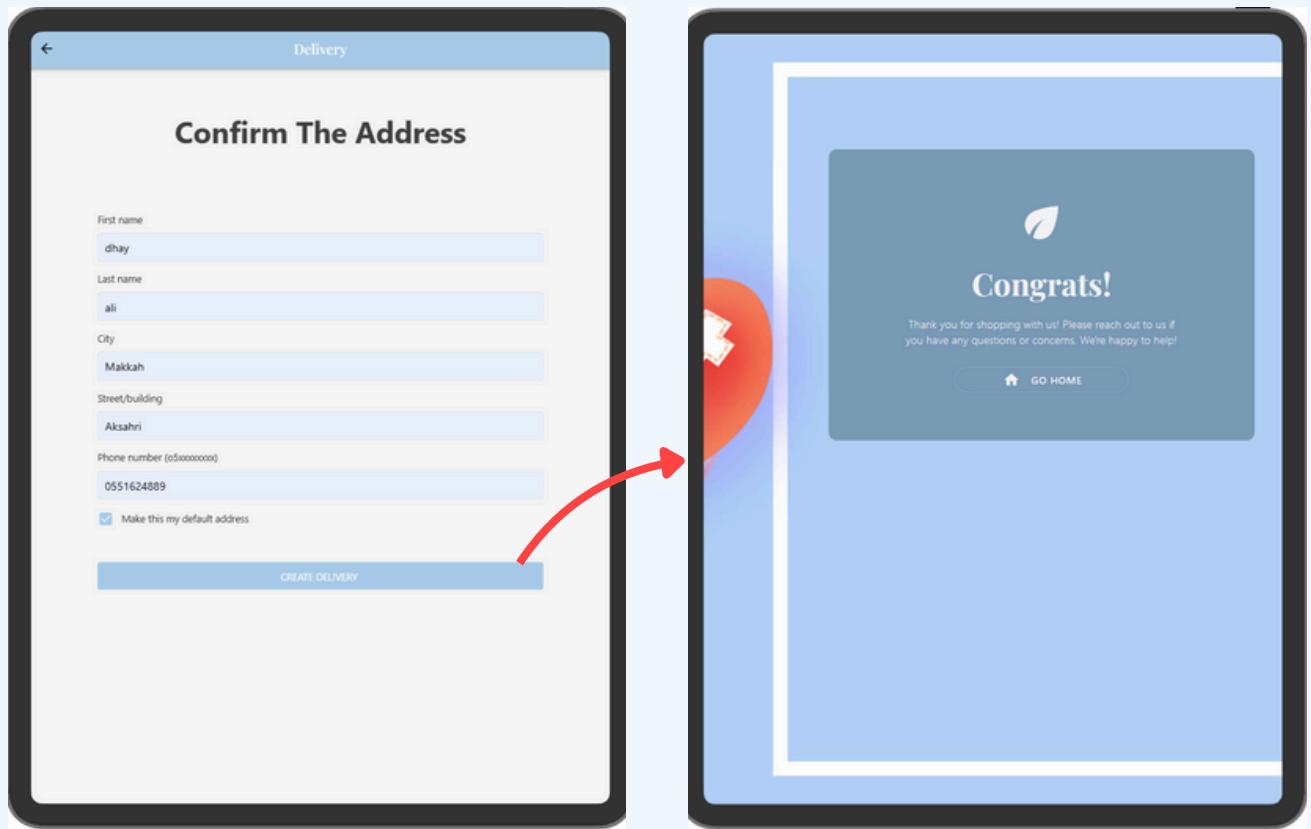
# Chapter 4



## Testing by Dhay

### 2. Create a delivery

**Description:** Ensure that the system will add the delivery information to the database and display a page with a confirmation message



# Chapter 4



## Testing by Jana

### 1. Empty entering technical support form

**Description:** Ensure that an error message appears to the user when the field is left empty.

The image shows a mobile application interface titled "Technical support Form". The screen displays a message: "Please Fill out the form to solve your problem or inquiry send". Below this message are three input fields: "Name", "Email", and "Problem or Inquiry", each with a placeholder text ("Enter name...", "Enter email...", "Enter problem or inquiry...") and a red border. Red error messages "This field is required." are positioned below each of these input fields. At the bottom of the screen is a blue button labeled "CREATE FORM".

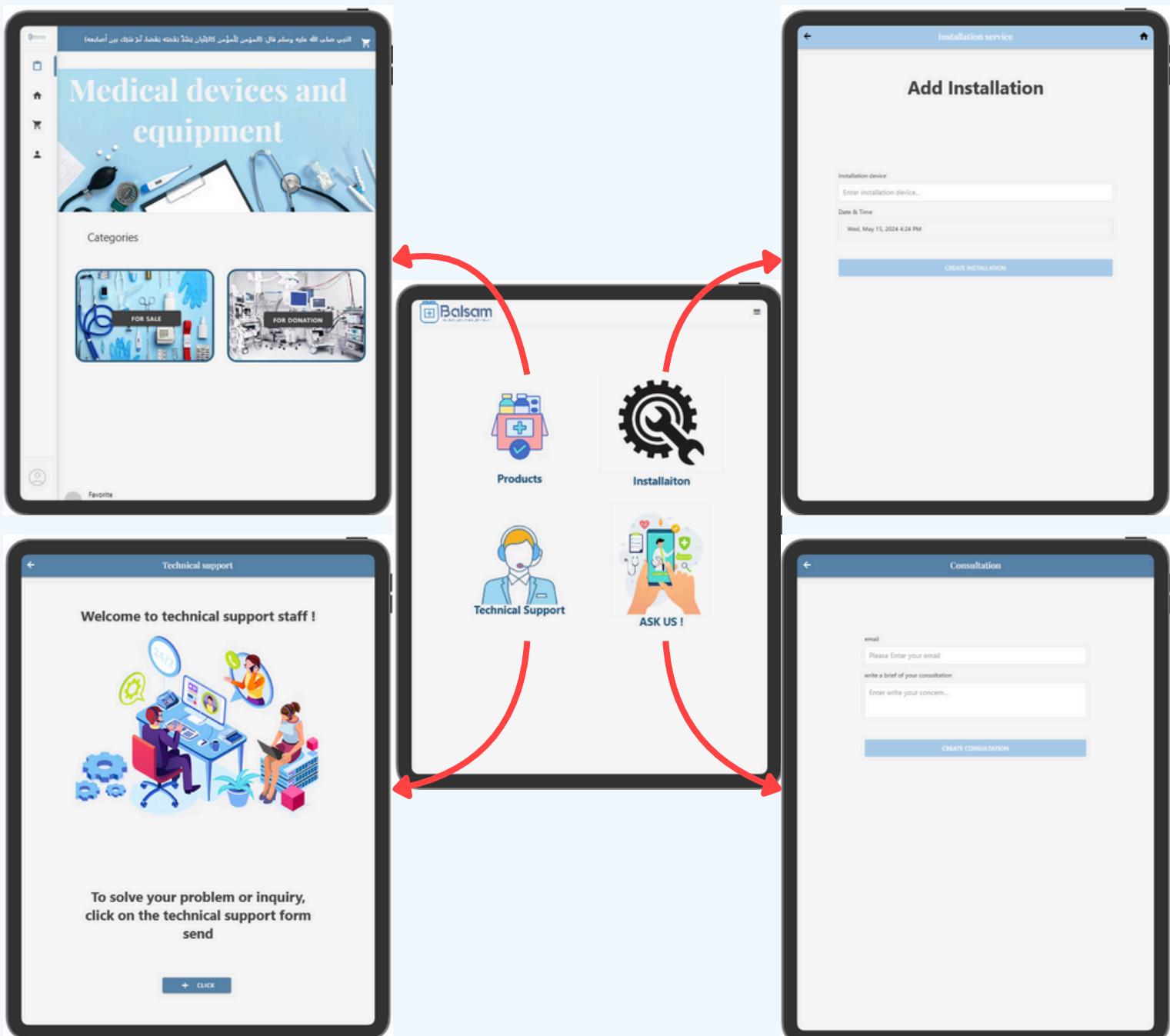
# Chapter 4



## Testing by Jana

### 2. Access to all services

**Description:** Successfully reached the intended screen.



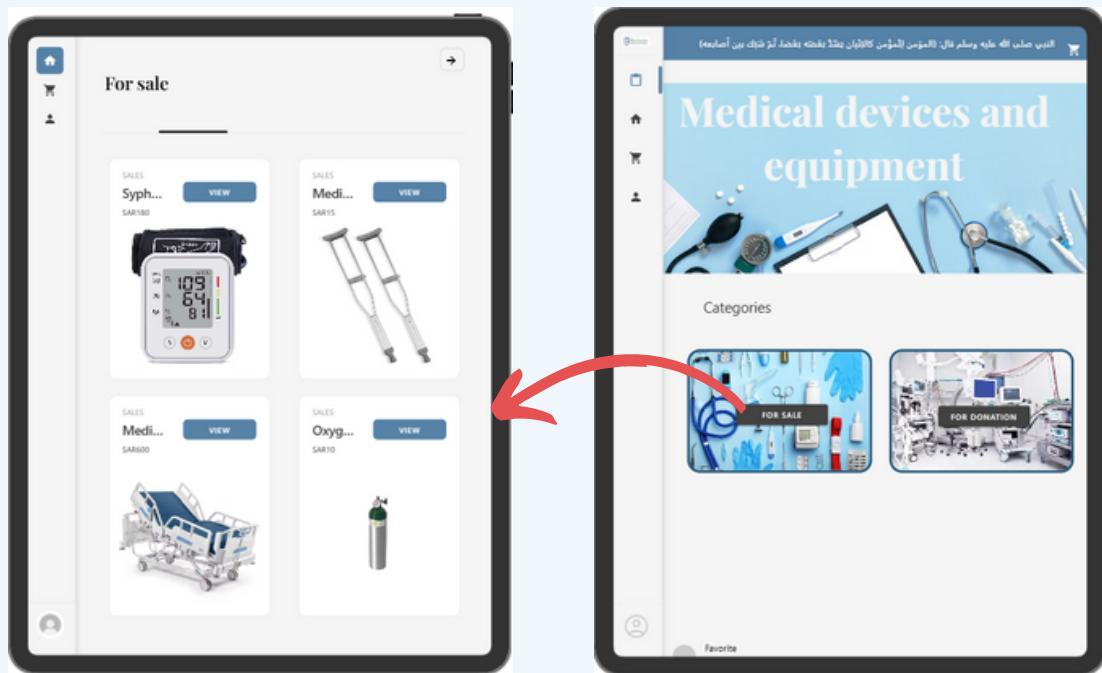
# Chapter 4



## Testing by Ghadah

### 1. access the product for sale screen

**Description:** successfully moving to the intended screen by clicking on either the image or the button



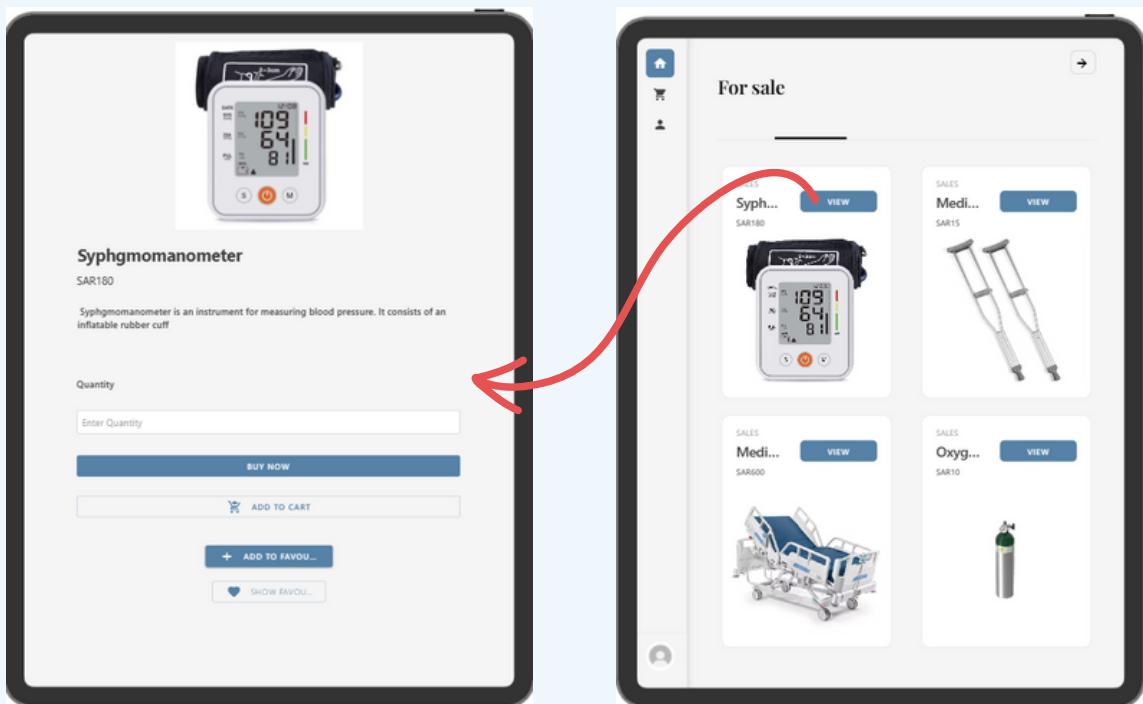
# Chapter 4



## Testing by Ghadah

### 2. access the product detail screen

**Description:** successfully moving to the specified product screen by clicking on -view- button



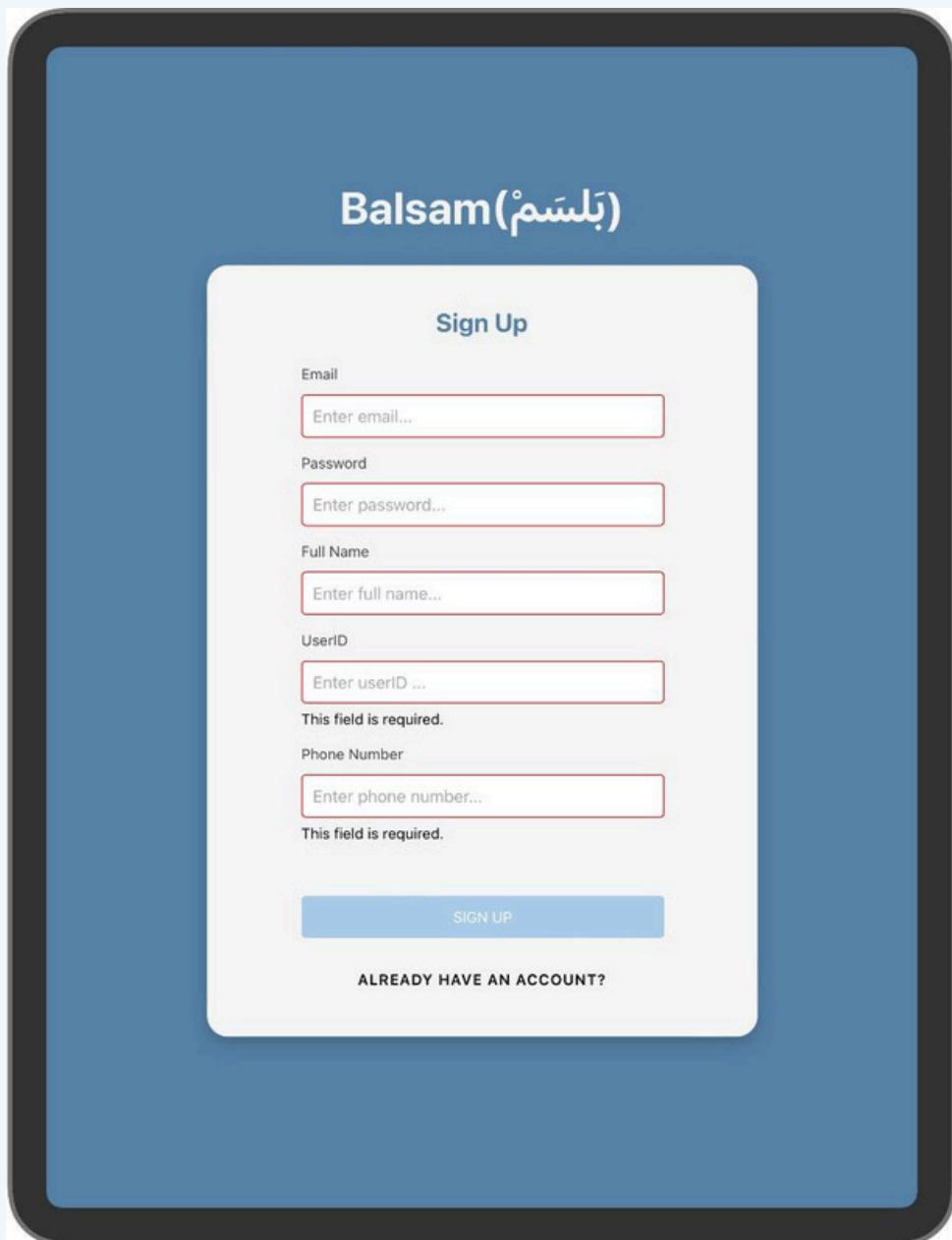
# Chapter 4



## Testing by Khadeejh

### 1. Empty field in create account

**Description:** If the user does not enter his information, an error message will appear on the screen



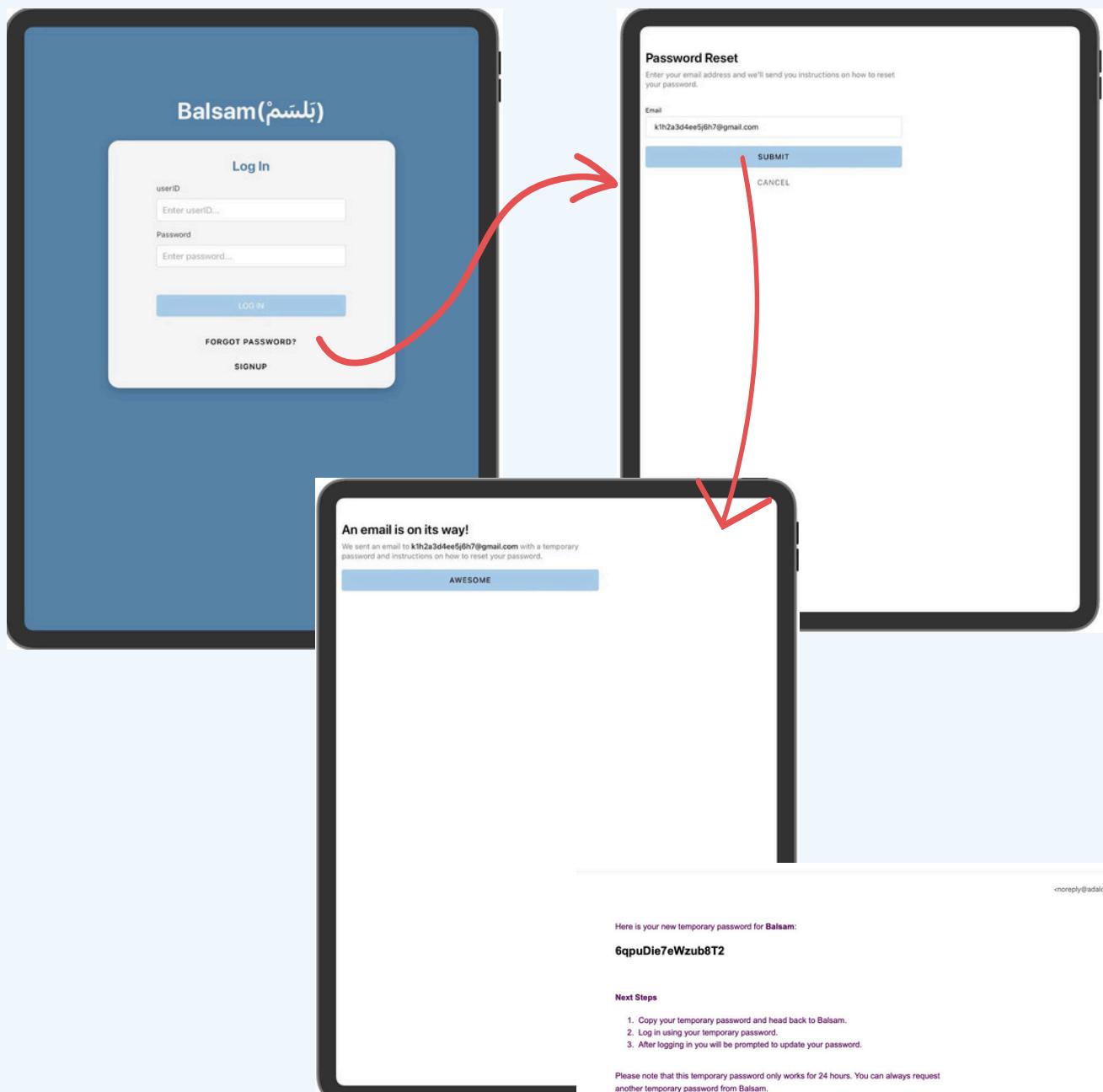
# Chapter 4



## Testing by Khadeejh

### 2. Reset password

**description:** If the user forgets his password, he can reset it via email and the message will be delivered to the email registered in Database.



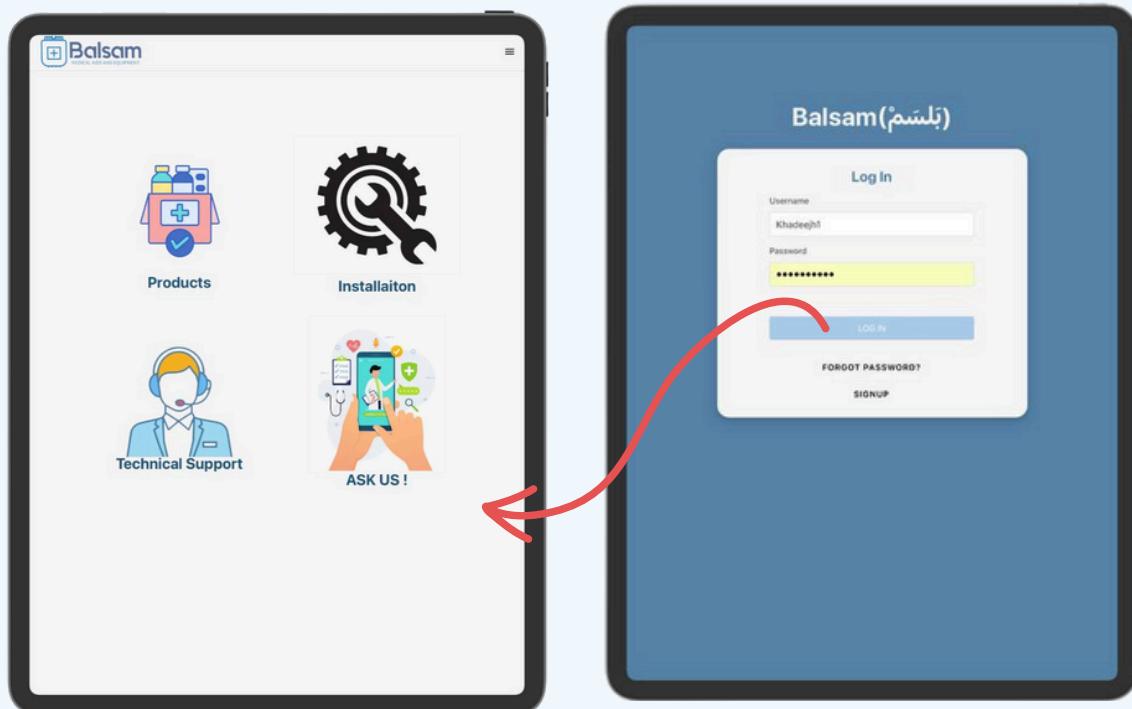
# Chapter 4



## Testing by Shaima

### 1. Log in to an existing account

**description:** successfully moving to the service screen by clicking on -login- button



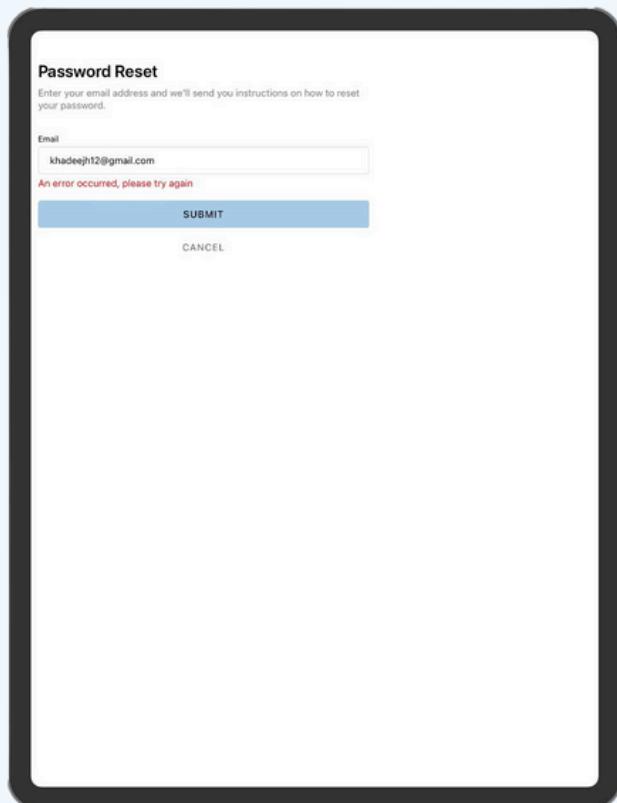
# Chapter 4

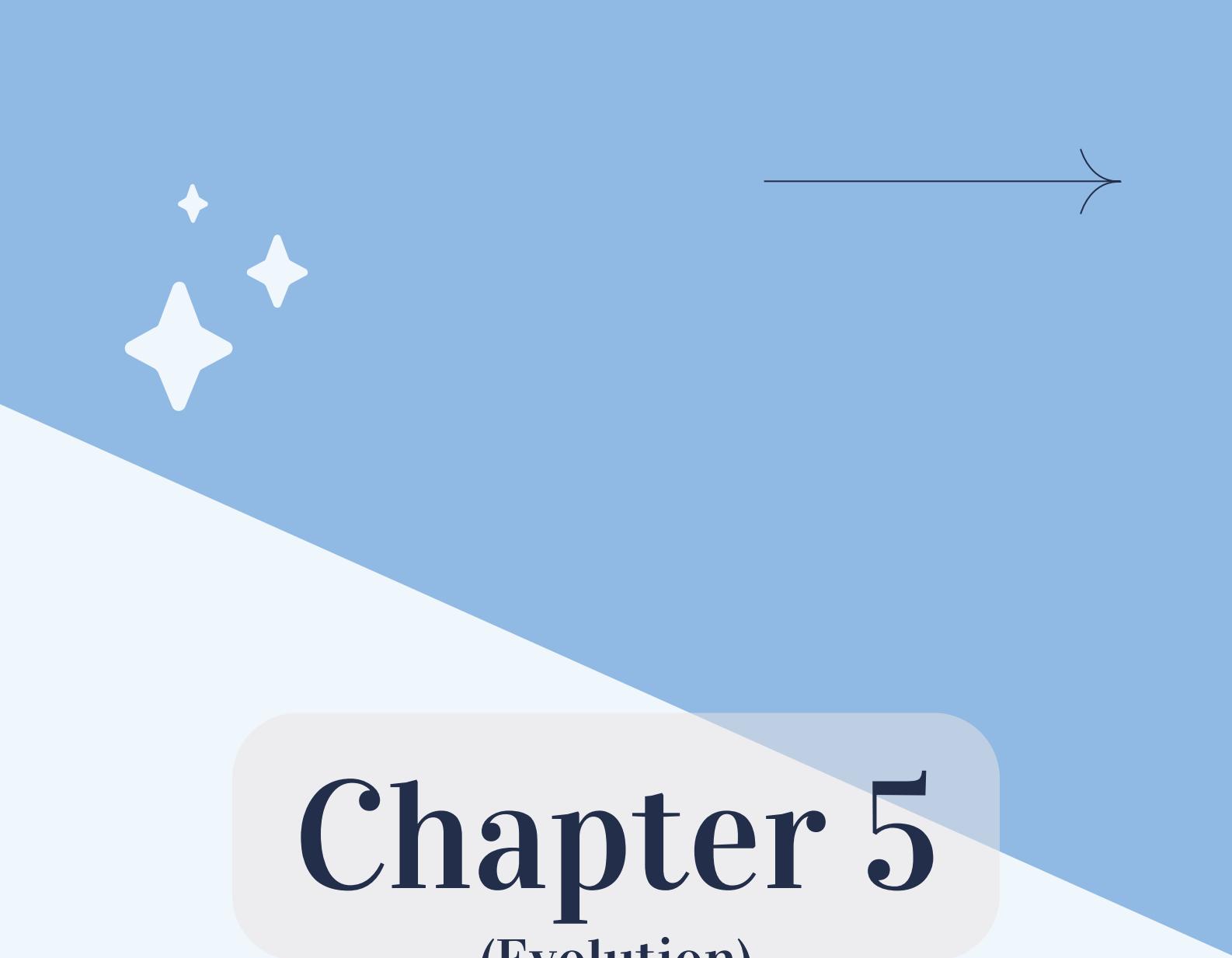


## Testing by Shaima

### 2. Write the email wrong in the reset password process

**description:** After clicking the "Submit" button, an error message will appear to indicate that the action was unsuccessful.





# Chapter 5

(Evolution)



# Chapter 5



## Evolution by Jana

### Home page (services)

Description : The code is well commented, meaning it includes explanatory notes to clarify the purpose of different sections, functions, or variables. In addition, it is designed in a way that separates HTML, which defines the structure of a web page, and CSS, which defines the style and appearance of HTML elements.

The image shows a code editor with two tabs: "Home.html" and "styles.css".

**Home.html:**

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Home Page</title>
    <link rel="stylesheet" href="styles.css"> <!-- Link to your external CSS file -->
</head>
<body>
    <header class="toolbar">
        <!-- Container for the toolbar -->
        <div class="logo">
            <a href="login.html">
                
            </a>
        </div>
    </header>
    <main class="content">
        <div class="image-grid">
            <!-- Each image links to a different page -->
            <div class="image-item">
                <a href="page1.html">
                    
                    <h3>Products</h3>
                </a>
            </div>
            <div class="image-item">
                <a href="installation.html">
                    
                    <h3>Installation</h3>
                </a>
            </div>
            <div class="image-item">
                <a href="page3.html">
                    
                    <h3>Technical Support</h3>
                </a>
            </div>
            <div class="image-item">
                <a href="Consultation.html">
                    
                    <h3>Ask US!</h3>
                </a>
            </div>
        </div>
    </main>
</body>
</html>
```

**styles.css:**

```
/* Reset default margin and padding */
body, html {
    margin: 0;
    padding: 0;
}

/* Basic styling */
.toolbar {
    background-color: #fffffe; /* White background for the toolbar */
    padding: 10px 0;
    display: flex;
    justify-content: space-between; /* Align logo and nav links to opposite ends */
    align-items: center; /* Center vertically */
    border-bottom: 2px solid #ccc; /* Gray border (underside) */
}

.toolbar .logo img {
    height: 80px; /* Adjust logo height as needed */
    padding-left: 20px; /* Add padding to separate logo from nav links */
}

.toolbar .nav {
    text-align: center;
}

.toolbar .nav ul {
    list-style: none;
    padding: 0;
}

.toolbar .nav ul li {
    display: inline;
    margin-right: 20px;
}

.toolbar .nav ul li a {
    color: #5382a2; /* Dark blue color for nav links */
    text-decoration: none;
    font-weight: bold;
}

.toolbar .nav ul li a .content {
    max-width: 1200px;
    margin: 0 auto;
    padding: 20px;
}

.image-grid {
    display: flex;
    justify-content: space-around;
    flex-wrap: wrap;
}

.image-item {
    text-align: center;
    margin-bottom: 20px;
    width: 45%; /* Set width to fit two items in a row */
}
```

# Chapter 5

## Evolution by Dhay

### Consultation page

#### Description :

- I added more comments to make it easier for the code developers
- Move style to an external CSS file

The image shows a code editor with two tabs: 'Consultation.html' and 'conStyle.css'. The 'Consultation.html' tab contains the following HTML code:

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1.0">
6     <title>Consultation</title>
7     <link rel="stylesheet" href="conStyle.css"> <!-- Linking to an external CSS file for styling -->
8   </head>
9   <body>
10    <div class="toolbar">
11      <h1>Consultation Form</h1>
12    </div>
13    <div class="container">
14      <form action="#" method="post">
15        <label for="email">Your Email:</label>
16        <input type="text" id="email" name="email" required> <!-- Input field for the user's email, marked as required -->
17
18        <label for="consultation">Brief Description of Consultation:</label>
19        <textarea id="consultation" name="consultation" required></textarea> <!-- Textarea for the user to provide a brief description-->
20
21        <input type="submit" value="Create a Consultation"> <!-- button to create the consultation -->
22      </form>
23    </div>
24  </body>
25 </html>
```

The 'conStyle.css' tab contains the following CSS code:

```
1 body {
2   margin: 0;
3   padding: 0;
4   font-family: Arial, sans-serif;
5   background-color: #f5f5f5; /* Light gray background */
6 }
7 .toolbar {
8   background-color: #5382a2;
9   color: white;
10  padding: 10px 20px;
11  display: flex;
12  justify-content: center; /* Center the text horizontally */
13  align-items: center;
14 }
15 .toolbar h1 {
16   margin: 0;
17 }
18 .container {
19   margin: 10px auto;
20   width: 600px;
21   max-width: 600px;
22   background-color: white;
23   padding: 20px;
24   border-radius: 10px;
25   box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);
26 }
27 h2 {
28   text-align: center;
29   color: #5382a2;
30   display: none; /* Hide the heading */
31 }
32 label {
33   font-size: 1em;
34   color: #333;
35   display: block;
36   margin-bottom: 5px;
37 }
38 input[type="text"] {
39   width: 100%;
40   padding: 10px;
41   margin-bottom: 15px;
42   border: 1px solid #ccc;
43   border-radius: 4px;
44   box-sizing: border-box;
45 }
46 textarea {
47   resize: vertical;
48   min-height: 100px;
49 }
50 input[type="submit"] {
51   background-color: #5382a2;
52   color: white;
53   padding: 10px 20px;
54   border: none;
55   border-radius: 4px;
56   cursor: pointer;
57   font-size: 1em;
58 }
59 input[type="submit"]:hover {
60   background-color: #805693;
61 }
```

# Chapter 5

## Evolution by Khadeejah

### Login page

Description: I've annotated the code with clear comments explaining each section. Variable names like "userEmail" and "consultationDescription" are used for clarity. The CSS styling is moved to an external file for better organization.

The image shows three side-by-side code editor windows. The left window contains the HTML code for 'login.html'. The middle window contains the CSS code for 'signup\_login\_style.css'. The right window shows the continuation of the CSS code from the middle window. The code is annotated with comments explaining the structure and styling.

```
login_balsam.html
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1.0">
6     <!-- Title of the page -->
7     <title>Login</title>
8     <!-- External stylesheet for styling -->
9     <link rel="stylesheet" href="signup_login_style.css">
10    </head>
11  <body>
12
13  <!-- Main content container -->
14  <div class="container">
15    <!-- Heading for the login section -->
16    <h1>BalSam<sup>اے سام</sup></h1>
17
18    <!-- Form for user login -->
19    <form action="Home.html" method="post">
20      <!-- Username input field with label -->
21      <label for="username">Username:</label>
22      <input type="text" placeholder="Enter Username...." name="username" required>
23      <!-- Password input field with label -->
24      <label for="password">Password:</label>
25      <input type="password" placeholder="Enter Password...." name="password" required>
26      <!-- Submit button for logging in -->
27      <input type="submit" value="Log In">
28
29    </form>
30    <!-- Links for forgotten password and sign up -->
31    <div class="button">
32      <a href="#">FORGET PASSWORD ?<br></a>
33      <a href="SignUp.html">SIGN UP </a>
34    </div>
35
36  </div>
37
38  </body>
39
40  </html>
```

```
signup_login_style.css
1 body {
2   margin: 0;
3   padding: 0;
4   font-family: Arial, sans-serif;
5   background-color: #5382a2; /* Blue background */
6   display: flex;
7   justify-content: center;
8   align-items: center;
9   height: 100vh;
10  flex-direction: column;
11 }
12 /* Styles for the container element */
13 .container {
14   width: 300px;
15   background-color: white;
16   padding: 20px;
17   border-radius: 10px;
18   box-shadow: 0 0 10px rgba(0, 0, 0, 0.5);
19 }
20 /* Styles for the main heading */
21 h1 {
22   text-align: center;
23   color: white;
24   margin-bottom: 20px;
25 }
26 /* Styles for the secondary heading */
27 h2 {
28   text-align: center;
29   color: #5382a2;
30 }
31 /* Styles for text input fields and submit button */
32 input[type="text"],
33 input[type="password"],
34 input[type="email"],
35 input[type="tel"],
36 input[type="submit"] {
37   width: 100%;
38   padding: 10px;
39   margin: 8px 0;
40   border: 1px solid #ccc;
41   border-radius: 4px;
42   box-sizing: border-box;
43 }
44 /* Styles for the submit button */
45 input[type="submit"] {
46   background-color: #a9cce9;
47   color: white;
48   cursor: pointer;
49 }
50 /* Hover styles for the submit button */
51 input[type="submit"]:hover {
52   background-color: #8085b3;
53 }
54 /* Styles for the button container */
55 .btns {
56   text-align: center;
57 }
58 /* Styles for links within the button container */
59 .btns a {
60   color: #222222;
61   text-decoration: none;
62   margin-right: 10px;
63   font-size: 10px;
64   font-family: Lucida Console;
65   font-weight: bold;
66 }
67 /* Hover styles for links within the button container */
68 .btns a:hover {
69   text-decoration: underline;
70 }
71 /* Styles for labels */
72 label {
73   font-size: 12px;
74   color: #333333;
75   display: block;
76   margin-bottom: 5px;
77 }
```

# Chapter 5

## Evolution By Shaima

### Sign up page

Description: I've added descriptive comments throughout the code to explain its functionality, and I've made the variable names clearer. Additionally, I've moved the CSS styling to an external file for better organization and maintenance.

The image shows three side-by-side Sublime Text windows illustrating the evolution of a web application's codebase:

- Left Window:** A CSS file named `signup_login_style.css`. It contains styles for a main container, input fields, and a submit button, including colors (#5382a2, #a9cce9), font families (Arial, sans-serif), and various border and padding properties.
- Middle Window:** A CSS file named `signup_login_style.css`. This version includes additional comments explaining the purpose of each section: /\* Styles for the container element \*/ and /\* Styles for the submit button \*/.
- Right Window:** An HTML file named `Home.html`. It shows the full structure: DOCTYPE, head (meta charset, viewport, title), and body (header with logo and links, main content grid with image items).

# Chapter 5

## Evolution By Ghadah

### Installation page

Description : I hav added descriptive comments to explain the code purpose and functionaltiy, changung variable names to more clearly ones. As well seperating the styling to an external CSS file.

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  <link rel="stylesheet" href="style.css">
5  <meta charset="UTF-8">
6  <meta name="viewport" content="width=device-width, initial-scale=1.0">
7  <title>Add installation</title>
8  <link rel="stylesheet" href="style.css">
9  </head>
10 <body>
11  <!---class of the top navigation menu-->
12  <div class="topnav">
13  <a class="active" href="#home"><</a>
14  <a>Add Installation</a>
15  <a href="#about">Home</a>
16
17 </div>
18 <h2>Add Installation</h2>
19 <div class="container">
20  <form action="#" method="post">
21  <label>installation device</label>
22  <input type="text" placeholder="Enter installation device" name="installationDevice" required>
23  <label>Date & Time:</label>
24  <input type="date" placeholder="Enter Email">
25  <input type="submit" value="CREATE INSTALLATION">
26 </form>
27 <div class="button">
28  </a>
29 </div>
30 </div>
31 </body>
32 </html>
```

```
1 /*styling the background and page*/
2 body {
3  padding: 0;
4  font-family: Arial, sans-serif;
5  background-color: #f6f6f6; /* grey background */
6  justify-content: center;
7  align-items: center;
8  height: 100vh;
9  flex-direction: column;
10 }
11
12 }
13 h1 {
14  text-align: center;
15  color: white;
16  margin-bottom: 20px;
17 }
18
19 h2 {
20  text-align: center;
21  margin-bottom: 70px;
22  color: black;
23 }
24 input[type="text"],
25 input[type="password"],
26 input[type="date"],
27 input[type="submit"] {
28  width: 70%;
29  height: 70px;
30  padding: 10px;
31  margin: 8px 0px 8px 10px;
32  display: block;
33  border: 0.2px solid #ccc;
34  border-radius: 0px;
35  box-sizing: border-box;
36  margin-left: 50px;
37  justify-content: left;
38  align-items: left;
39  display: flex;
40 }
41 input[type="submit"] {
42  background-color: #9ccef9;
43  margin-top: 30px;
44 }
```

# THANKS!

In conclusion, the Balsam Project is a friendly and innovative initiative that's here to make a positive impact on the healthcare industry. With its mission to improve accessibility and user experience, Balsam is like a helpful friend who's always there to lend a hand.

In a world where healthcare can sometimes feel overwhelming, the Balsam Project is here to bring convenience, accessibility, and a warm, friendly atmosphere. So jump on board, join the Balsam community, and let's make healthcare a friendlier and more enjoyable experience for everyone.

and we hope in the future will be no more worrying about limited access to medical devices! with Balsam